BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 70h-7Fh Processors

© 2016 Advanced Micro Devices, Inc. All rights reserved.

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

Trademarks

AMD, the AMD Arrow logo, AMD Virtualization and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

ARM is a registered trademarks of ARM Limited.

HyperTransport is a licensed trademark of the HyperTransport Technology Consortium.

Microsoft Windows and DirectX are registered trademarks of Microsoft Corporation.

MMX is a trademark of Intel Corporation.

PCI Express, PCI-X and PCIe are registered trademarks of PCI-Special Interest Group (PCI-SIG).

OpenCL is a trademark of Apple Inc., and is used under license by Khronos.

DivX is a registered trademark of DivX, LLC.

Dolby Laboratories, Inc.

Manufactured under license from Dolby Laboratories.

Rovi Corporation

This device is protected by U.S. patents and other intellectual property rights. The use of Rovi Corporation's copy protection technology in the device must be authorized by Rovi Corporation and is intended for home and other limited pay-per-view uses only, unless otherwise authorized in writing by Rovi Corporation.

Reverse engineering or disassembly is prohibited.

USE OF THIS PRODUCT IN ANY MANNER THAT COMPLIES WITH THE MPEG ACTUAL OR DE FACTO VIDEO AND/OR AUDIO STANDARDS IS EXPRESSLY PROHIBITED WITHOUT ALL NECESSARY LICENSES UNDER APPLICABLE PATENTS. SUCH LICENSES MAY BE ACQUIRED FROM VARIOUS THIRD PARTIES INCLUDING, BUT NOT LIMITED TO, IN THE MPEG PATENT PORTFOLIO, WHICH LICENSE IS AVAILABLE FROM MPEG LA, L.L.C., 6312 S. FIDDLERS GREEN CIRCLE, SUITE 400E, GREENWOOD VILLAGE, COLORADO 80111.

Table of Contents

1 Overview 21
1.1 Intended Audience 21
1.2 Reference Documents 21
1.3 Conventions 21
1.3.1 Numbering 21
1.3.2 Arithmetic And Logical Operators 22
1.3.3 Operator Precedence and Associativity 23
1.4 Definitions. 23
1.5 Changes Between Revisions and Product Variations 28
1.5.1 Revision Conventions 28
1.5.2 Major Changes 29
1.5.2.1 Major Changes to Core/NB Performance Counters 29

2 Functional Description 29
2.1 Processor Overview 29
2.2 System Overview 32
2.3 Processor Initialization 32
2.3.1 BSC Initialization. 32
2.3.2 AP Initialization 33
2.3.3 Using L2 Cache as General Storage During Boot 33
2.4 Core. 35
2.4.1 Compute Unit. 35
2.4.2 Caches and TLBs 35
2.4.2.1 Registers Shared by Cores in a Compute Unit 36
2.4.3 Virtual Address Space 36
2.4.4 Processor Cores and Downcoring 36
2.4.4.1 Software Downcoring using D18F3x190[DisCore] 37
2.4.5 Physical Address Space 37
2.4.6 System Address Map 38
2.4.6.1 Memory Access to the Physical Address Space 38
2.4.6.1.1 Determining Memory Type 38
2.4.6.1.2 Determining The Access Destination for Core Accesses 38
2.4.7 Timers 39
2.4.8 Implicit Conditions for TLB Invalidation 39
2.4.9 Interrupts 39
2.4.9.1 Local APIC 39
2.4.9.1.1 Detecting and Enabling 40
2.4.9.1.2 APIC Register Space 40
2.4.9.1.3 ApicId Enumeration Requirements 40
2.4.9.1.4 Physical Destination Mode 41
2.4.9.1.5 Logical Destination Mode 41
2.4.9.1.6 Interrupt Delivery 41
2.4.9.1.7 Vectored Interrupt Handling 41
2.4.9.1.8 Interrupt Masking 42
2.4.9.1.9 Spurious Interrupts 42
2.4.9.1.10 Spurious Interrupts Caused by Timer Tick Interrupt 42
2.4.9.1.11 Lowest-Priority Interrupt Arbitration 42
2.4.9.1.12 Inter-Processor Interrupts 43

2.4.9.1.13 APIC Timer Operation 43
2.4.9.1.14 Generalized Local Vector Table 43
2.4.9.1.15 State at Reset 43
2.4.9.2 System Management Mode (SMM) 43
2.4.9.2.1 SMM Overview 44
2.4.9.2.2 Operating Mode and Default Register Values 44
2.4.9.2.3 SMI Sources And Delivery 44
2.4.9.2.4 SMM Initial State 45
2.4.9.2.5 SMM Save State 45
2.4.9.2.6 Exceptions and Interrupts in SMM 51
2.4.9.2.7 The Protected ASeg and TSeg Areas 51
2.4.9.2.8 SMM Special Cycles 51
2.4.9.2.9 Locking SMM 51
2.4.9.2.10 Synchronizing SMM Entry (Spring-Boarding) 51
2.4.10 Secure Virtual Machine Mode (SVM). 53
2.4.10.1 BIOS support for SVM Disable 53
2.4.11 CPUID Instruction 54
2.4.11.1 Multi-Core Support 54
2.5 Power Management. 55
2.5.1 Processor Power Planes And Voltage Control. 55
2.5.1.1 Serial VID Interface 55
2.5.1.1.1 SVI2 Features 55
2.5.1.2 Internal VID Registers and Encodings 56
2.5.1.2.1 MinVid and MaxVid Check 56
2.5.1.3 Low Power Features 56
2.5.1.3.1 PSIx_L Bit 56
2.5.1.3.1.1 BIOS Requirements for PSI0_L 56
2.5.1.4 Voltage Transitions 57
2.5.2 CPU Core Power Management 58
2.5.2.1 Core P-states 58
2.5.2.1.1 Core P-state Naming and Numbering 58
2.5.2.1.1.1 Software P-state Numbering 58
2.5.2.1.1.2 Hardware P-state Numbering 59
2.5.2.1.2 Core P-state Control 59
2.5.2.1.3 Core P-state Visibility 59
2.5.2.1.4 Core P-state Limits 59
2.5.2.1.5 BIOS Requirements for Core P-state Initialization and Transitions . . . 60
2.5.2.1.6 Processor-Systemboard Current Delivery Compatibility Check 60
2.5.2.1.7 BIOS COF and VID Requirements After Warm Reset 60
2.5.2.1.7.1 Core Maximum P-state Transition Sequence After Warm Reset 61
2.5.2.1.7.2 Core Minimum P-state Transition Sequence After Warm Reset 61
2.5.2.1.7.3 ACPI Processor P-state Objects 61
2.5.2.1.7.4 Fixed ACPI Description Table (FADT) Entries 63
2.5.2.1.7.5 XPSS (Microsoft® Extended PSS) Object 63
2.5.2.2 Core C-states 63
2.5.2.2.1 C-state Names and Numbers 63
2.5.2.2.2 C-state Request Interface 63
2.5.2.2.3 C-state Actions 64
2.5.2.2.3.1 C-state Probes and Cache Flushing 64
2.5.2.2.3.2 Core C1 (CC1) State 64
2.5.2.2.3.3 Core C6 (CC6) State 64

2.5.2.2.3.4 Package C6 (PC6) State 65
2.5.2.2.4 C-state Request Monitors 65
2.5.2.2.4.1 FCH Messaging 65
2.5.2.2.4.2 Cache Flush On Halt Saturation Counter 66
2.5.2.2.5 Exiting C-states 66
2.5.2.2.6 ACPI Processor C-state Objects 66
2.5.2.2.6.1 _CST 66
2.5.2.2.6.2 _CSD 66
2.5.2.2.6.3 _CRS 67
2.5.2.2.6.4 Fixed ACPI Description Table (FADT) Entries 67
2.5.2.2.7 BIOS Requirements for Initialization 67
2.5.2.3 Effective Frequency 67
2.5.3 NB Power Management 68
2.5.3.1 NB P-states 68
2.5.3.1.1 NB P-state Transitions 68
2.5.3.1.2 BIOS NB P-state Configuration 68
2.5.3.1.2.1 NB P-state COF and VID Synchronization After Warm Reset 68
2.5.3.1.2.2 NB P-state Transitions 69
2.5.3.2 NB C-states 69
2.5.4 Bandwidth Requirements 70
2.5.5 GPU Power Management 70
2.5.6 DRAM Power Management 71
2.5.6.1 Memory P-states 71
2.5.6.2 DRAM Self-Refresh 71
2.5.6.3 Stutter Mode 72
2.5.6.4 EVENT_L 72
2.5.7 System Power Management States 72
2.5.7.1 S-states 72
2.5.7.1.1 ACPI Suspend to RAM State (S3) 72
2.5.8 Application Power Management (APM) 72
2.5.8.1 Core Performance Boost (CPB) 73
2.5.8.1.1 C-state Boost 73
2.5.8.2 Thermal Limiting 73
2.5.8.3 Bidirectional Application Power Management (BAPM) 73
2.6 Performance Monitoring 74
2.6.1 Performance Monitor Counters 74
2.6.1.1 Core Performance Monitor Counters 74
2.6.1.2 NB Performance Monitor Counters 75
2.6.2 Instruction Based Sampling (IBS) 76
2.7 Configuration Space 77
2.7.1 MMIO Configuration Coding Requirements 77
2.7.2 MMIO Configuration Ordering 77
2.7.3 Processor Configuration Space 78
2.8 Northbridge (NB) 78
2.8.1 NB Architecture 78
2.8.2 NB Routing 78
2.8.2.1 Address Space Routing 78
2.8.2.1.1 DRAM and MMIO Memory Space 78
2.8.2.1.2 IO Space 79
2.8.2.1.3 Configuration Space 79
2.8.2.1.3.1 Recommended Buffer Count Settings Overview 80

2.9 DRAM Controllers (DCTs) 81
2.9.1 Common DCT Definitions. 81
2.9.2 DCT Frequency Support 82
2.9.3 DCT Configuration Registers 85
2.9.4 DDR Pad to Processor Pin Mapping 85
2.9.4.1 DDR Chip to Pad Mapping 86
2.9.5 DRAM Controller Direct Response Mode 89
2.9.6 DRAM Data Burst Mapping 89
2.9.7 SOC Specific Definitions. 90
2.9.8 PMU. 90
2.9.8.1 Upstream Mailbox1 Message Pending 90
2.9.8.2 Upstream Mailbox2 Message Pending 91
2.9.8.3 SRAM Message Block 91
2.9.9 DCT/DRAM Initialization and Resume 91
2.9.9.1 Low Voltage 92
2.9.9.2 DDR Phy Initialization 92
2.9.9.2.1 Phy General Configuration 93
2.9.9.2.2 Phy Voltage Level Programming 93
2.9.9.2.3 DRAM Channel Frequency 94
2.9.9.2.4 DRAM Address, Command, and Output Driver Control 94
2.9.9.2.5 DRAM Data Bus Configuration 104
2.9.9.2.6 Phy FIFO Configuration 114
2.9.9.2.7 Phy Predriver Initialization. 115
2.9.9.2.8 Phy Auto-Calibration 119
2.9.9.2.8.1 One-Time Pre-PMU Calibration 119
2.9.9.2.8.2 Fence CalOnce 119
2.9.9.2.8.3 Auto Calibration 119
2.9.9.2.9 PMU Firmware Load 119
2.9.9.2.10 Phy Registers Required for S3 Resume 120
2.9.9.2.11 Calculating Round Trip Command Delays 120
2.9.9.3 SPD ROM-Based Configuration 120
2.9.9.3.1 DRAM ODT Pin Control 121
2.9.9.4 DCT Specific Configuration 122
2.9.9.4.1 DDR Turnaround Parameters 123
2.9.9.4.1.1 TrdrdBan, TrdrdSdSc, TrdrdSdDc, and TrdrdDd (Rd->Rd Timing) 123
2.9.9.4.1.2 TwrwrSdSc, TwrwrSdDc, TwrwrDd (Wr->Wr Timing) 123
2.9.9.4.1.3 Twrrd (Write to Read DIMM Termination Turn-around) 124
2.9.9.4.1.4 TrwtTO (Read-to-Write Turnaround for Data, DQS Contention) . . 124
2.9.9.5 DRAM Device Initialization and Training 124
2.9.9.6 DRAM Training 125
2.9.9.6.1 Training MaxRdLatency 125
2.9.9.7 Synchronous Channel Initialization 125
2.9.9.8 DRAM Channel Disable 125
2.9.9.9 DRAM Phy Power Savings 126
2.9.10 Continuous Pattern Generation 127
2.9.10.1 DCT Training Pattern Generation 127
2.9.10.1.1 Activate and Precharge Command Generation 127
2.9.10.1.2 Read and Write Command Generation 128
2.9.10.1.3 Configurable Data Patterns 129
2.9.10.1.3.1 Static Data Pattern Override 130
2.9.10.1.3.2 Xor Data Pattern Override 131

2.9.10.1.4 Data Comparison 131
2.9.10.1.4.1 Activate and Precharge Traffic Generation 132
2.9.10.1.5 BubbleCnt and CmdStreamLen Programming 132
2.9.11 Memory Interleaving Modes 133
2.9.11.1 Chip Select Interleaving 134
2.9.12 Memory Hoisting 137
2.9.12.1 DramHoleOffset Programming 137
2.9.13 DRAM CC6/PC6 Storage 139
2.9.13.1 Fixed Storage 139
2.9.14 DRAM On DIMM Thermal Management and Power Capping 140
2.10 Thermal Functions 142
2.10.1 The Tctl Temperature Scale 142
2.10.2 Temperature Slew Rate Control 143
2.10.3 Temperature-Driven Logic 143
2.10.3.1 PROCHOT_L and Hardware Thermal Control (HTC) 143
2.10.3.2 Software P-state Limit Control 144
2.10.3.3 THERMTRIP 144
2.11 Root Complex 146
2.11.1 Overview 146
2.11.2 Interrupt Routing 146
2.11.2.1 IOAPIC Configuration 147
2.11.3 Links 148
2.11.3.1 Overview 148
2.11.3.2 Link Configurations 148
2.11.4 Root Complex Configuration 148
2.11.4.1 LPC MMIO Requirements 148
2.11.4.2 Configuration for non-FCH Bridges 149
2.11.4.3 Link Configuration and Initialization 149
2.11.4.3.1 Link Configuration and Core Initialization 149
2.11.4.3.2 Link Training 149
2.11.4.4 Miscellaneous Features 149
2.11.4.4.1 Lane Reversal 149
2.11.4.4.2 Link Speed Changes 149
2.11.4.4.2.1 Autonomous Link Speed Changes 150
2.11.4.4.3 Deemphasis 150
2.11.4.5 Link Power Management 150
2.11.4.5.1 Link States 150
2.11.4.5.2 Dynamic Link-width Control 150
2.11.4.6 Link Test and Debug Features 150
2.11.4.6.1 Compliance Mode 150
2.11.5 FCH Messages 151
2.11.6 BIOS Timer 151
2.11.7 PCIe® Client Interface Control 151
2.12 IOMMU 153
2.12.1 IOMMU Configuration Space 153
2.12.2 IOMMU Initialization 153
2.12.2.1 IOMMU L1 Initialization 153
2.12.2.2 IOMMU L2 Initialization 154
2.12.2.3 IOMMU SMI Filtering 154
2.12.2.4 IOMMU Power Gating 155
2.12.2.4.1 Static Power Down IOMMU L2 155

2.12.2.4.2 Memory Power Gating in IOMMU L1 155
2.13 System Management Unit (SMU). 155
2.13.1 System Management Controller (SMC) 155
2.14 Graphics Processor (GPU) 155
2.14.1 Graphics Memory Controller (GMC) 155
2.14.2 Frame Buffer (FB) 155
2.14.2.1 General Guidelines 155
2.14.2.2 Host Data Path Guidelines 156
2.15 Audio Processor (ACG) 156
2.15.1 Programming ACG Host Address Mapping 156
2.15.2 Programming ACG DMA Apertures 157
2.16 RAS Features 158
2.16.1 Machine Check Architecture 158
2.16.1.1 Machine Check Registers 158
2.16.1.2 Machine Check Errors 159
2.16.1.3 Error Detection, Action, Logging, and Reporting 160
2.16.1.3.1 MCA Conditions that Cause Shutdown 161
2.16.1.3.2 Error Logging During Overflow 161
2.16.1.4 MCA Initialization 162
2.16.1.5 Error Code 162
2.16.1.6 Handling Machine Check Exceptions 164
2.16.1.6.1 Differentiation Between System-Fatal and Process-Fatal Errors 166
2.16.1.7 Error Thresholding 167
2.16.1.8 Error Diagnosis 168
2.16.1.8.1 Common Diagnosis Information 169
2.16.1.9 Deferred Errors and Data Poisoning 170
2.17 FCH. 172
2.17.1 MMIO Programming for Legacy Devices. 172
2.17.2 USB Controllers. 172
2.17.2.1 USB Power Management 173
2.17.2.2 USB Interrupts 173
2.17.2.3 Enabling the xHCI Controller 174
2.17.2.4 USB2.0 Controller PHY Configuration and Calibration 174
2.17.2.5 USB2.0 Controller Common PHY Impedance Calibration 174
2.17.2.6 USB2.0 Controller ISO Device CRC False Error Detection 174
2.17.2.7 xHC USB2.0 Common PHY Calibration 175
2.17.2.8 xHC USB2.0 Port Drive Strength Adjustment 175
2.17.2.9 USB3.0 PHY Auto-Calibration Enablement 176
2.17.2.10 USB3.0 Port LFPS Differential Detection Threshold Adjustment 176
2.17.2.11 xHCI ISO Device CRC False Error Detection 176
2.17.2.12 xHCI PHY Clock Gating 176
2.17.2.13 xHCI Firmware Preload 176
2.17.2.14 xHCI Clear Pending PME on Sx State Entry 177
2.17.3 SATA 177
2.17.3.1 SATA Operating Mode 177
2.17.3.2 SATA Drive Detection in IDE Mode 177
2.17.3.3 SATA PHY Auto-Calibration Enablement 177
2.17.3.4 SATA PHY Fine Tuning 178
2.17.3.5 SATA PHY Reference Clock Selection 178
2.17.3.6 SATA Power Management 179
2.17.3.6.1 SATA PHY Power Saving 179

2.17.3.7 Enable Shadow Register Reload 179
2.17.3.8 SATA Interrupt Handling 179
2.17.3.8.1 Line Interrupt 179
2.17.3.8.2 MSI Message 179
2.17.3.9 Clear status of SATA PERR 180
2.17.4 LPC Bus Interface 180
2.17.4.1 Enabling LPC DMA Function 180
2.17.4.2 Enabling SPI 100 180
2.17.4.3 Enabling LPC_SMI Function 180
2.17.4.4 Serial Peripheral Interface (SPI) ROM 181
2.17.4.4.1 Enable SPI ROM Protection 181
2.17.4.4.2 Disable SPI ROM Protection 181
2.17.5 SD Controller 181
2.17.6 ASF Controller 181
2.17.7 Integrated Micro-Controller 182
2.17.8 On-Chip Clock Generator 182
2.17.8.1 Power Saving In Internal Clock Mode 182
2.17.8.2 Global A-Link / B-Link Clock Gating 182
2.17.8.3 CG_PLL CMOS Clock Driver Setting for Power Saving 183
2.17.9 FCH Gasket 183
2.17.10 A-Link Bridge 183
2.17.10.1 Detection of Upstream Interrupts 183
2.17.10.2 AB Memory Power Saving 183
2.17.10.3 AB Internal Clock Gating 183
2.17.10.4 AB 32/64-Byte DMA Write Enable 183
2.17.11 Port Detection on XHCI Controller 184
2.17.12 AutoSize Function 184
2.17.13 FCH IP Address Mapping 184

3 Registers 185
3.1 Register Descriptions and Mnemonics 185
3.1.1 Northbridge MSRs In Multi-Core Products 189
3.1.2 Software Recommendation (BIOS, SBIOS) 189
3.1.3 See Keyword (See:) 190
3.1.4 Mapping Tables 190
3.1.4.1 Register Mapping 190
3.1.4.2 Index Mapping 190
3.1.4.3 Field Mapping 190
3.1.4.4 Broadcast Mapping 190
3.1.4.5 Reset Mapping 190
3.1.4.6 Valid Values 191
3.1.4.7 BIOS Recommendations 191
3.2 IO Space Registers 191
3.3 Device 0 Function 0 (Root Complex) Configuration Registers 193
3.4 Device 0 Function 2 (IOMMU) Configuration Registers 225
3.5 Device 1 Function 0 (Internal Graphics) Configuration Registers 264
3.6 Device 1 Function 1 (Audio Controller) Configuration Registers 276
3.7 Device 2 Function 0 (Host Bridge) Configuration Registers 287
3.8 Device 2 Function [5:1] (Root Port) Configuration Registers 288
3.9 HD Audio Controller 321
3.9.1 Device 9h Function 2 (Audio Controller) Configuration Registers 321

3.10 Device 18h Function 0 Configuration Registers 335
3.11 Device 18h Function 1 Configuration Registers 348
3.12 Device 18h Function 2 Configuration Registers 361
3.13 Device 18h Function 3 Configuration Registers 446
3.14 Device 18h Function 4 Configuration Registers 493
3.15 Device 18h Function 5 Configuration Registers 503
3.16 Northbridge IOAPIC Registers 517
3.17 IOMMU Memory Mapped Registers 519
3.18 APIC Registers 548
3.19 CPUID Instruction Registers 558
3.20 MSRs - MSR0000_xxxx 587
3.21 MSRs - MSRC000_0xxx 628
3.22 MSRs - MSRC001_0xxx 635
3.23 MSRs - MSRC001_1xxx 660
3.24 Core Performance Counter Events 679
3.24.1 PMCx0[1F:00] Events (FP) 679
3.24.2 PMCx0[3F:20] Events (LS) 680
3.24.3 PMCx0[5F:40] Events (DC) 682
3.24.4 PMCx[1:0][7F:60] Events (CU) 685
3.24.5 PMCx[1:0][9F:80] Events (IC) 690
3.24.6 PMCx[1,0][DF:C0] Events (EX, DE) 692
3.25 NB Performance Counter Events 697
3.25.1 PMCx0E[7:4] Events (Memory Controller) 698
3.25.2 PMCx0E[F:8] Events (Crossbar) 698
3.25.3 PMCx0F[F:0] Events (Crossbar) 701
3.25.4 NBPMCx1E[F:0] Events (Crossbar) 701
3.25.5 NBPMCx1F[F:0] Events (Memory Controller, Crossbar) 704
3.26 FCH Registers 707
3.26.1 Legacy Block Configuration Registers (IO) 707
3.26.2 AB Configuration Registers 728
3.26.3 SATA Controller 731
3.26.3.1 Device 11h Function 0 (SATA) Configuration Registers 731
3.26.3.2 SATA IO Registers 747
3.26.3.2.1 IDE Compatibility Mode and Native Mode Registers 747
3.26.3.2.2 IDE Bus Master Registers 749
3.26.3.3 SATA Memory Mapped AHCI Registers 750
3.26.3.3.1 Generic Host Control Registers 751
3.26.3.3.2 Port Control Registers 757
3.26.3.3.3 Enclosure Buffer Management Registers 771
3.26.4 USB Controllers 774
3.26.4.1 USB 2.0 (EHCI) 774
3.26.4.1.1 Devices 12h Function 0 (EHCI) Configuration Registers 774
3.26.4.1.2 EHCI Memory Mapped IO Registers 787
3.26.4.2 USB 3.0 (xHCI) 801
3.26.4.2.1 Device 10h Function 0 (xHCI) Configuration Registers 801
3.26.4.2.2 xHCI Memory Mapped IO Configuration Registers 819
3.26.4.2.2.1 USB xHCI Capability Registers 819
3.26.4.2.3 xHCI Power Management Registers 820
3.26.4.3 USB Legacy Registers 828
3.26.5 Secure Digital (SD) Controller 832
3.26.5.1 Device 14h Function 7 Configuration Registers (SD) 832

3.26.5.2 SD Host Controller Configuration Registers (SDHC) 837
3.26.6 SMBus Host Controller 853
3.26.6.1 Device 14h Function 0 (SMBus) Configuration Registers 853
3.26.6.2 ASF (Alert Standard Format) Registers 856
3.26.6.3 SMBus Registers 861
3.26.7 IOAPIC Registers 868
3.26.8 LPC-ISA Bridge. 870
3.26.8.1 Device 14h Function 3 (LPC Bridge) Configuration Registers 870
3.26.8.2 SPI Registers 883
3.26.8.3 eSPI Registers 892
3.26.9 High Precision Event Timer (HPET) Registers. 912
3.26.10 Miscellaneous (MISC) Registers 916
3.26.11 GPIO Pin control registers. 935
3.26.11.1 GPIO Registers 936
3.26.11.2 IOMux Registers 944
3.26.12 Power Management (PM) Registers 973
3.26.13 Power Management Block 2 (PM2) Registers. 1001
3.26.14 Standard ACPI Registers 1011
3.26.14.1 AcpiPmEvtBlk 1011
3.26.14.2 AcpiPm1CntBlk 1012
3.26.14.3 AcpiPm2CntBlk 1013
3.26.14.4 AcpiPmTmrBlk 1013
3.26.14.5 CpuCntBlk 1013
3.26.14.6 AcpiGpe0Blk 1014
3.26.14.7 SmiCmdBlk 1014
3.26.15 SMI Registers. 1015
3.26.16 Watchdog Timer (WDT) Registers 1040
3.26.17 Wake Alarm Device (AcDcTimer) Registers 1040
3.26.18 A-Link to AHB Bridge (AL2AHB) Configuration Registers 1044
3.26.19 DMA Registers 1049
3.26.20 I2C Configuration Registers 1064
3.26.21 UART Registers. 1080

4 Register List. 1092

List of Figures

Figure 1: A Compute Unit 31
Figure 2: A Processor 32
Figure 3: Memory Configuration with Memory Hole Inside of Region 138
Figure 4: Memory Configuration with Memory Hole Outside of Region 139
Figure 5: Tctl Scale 143
Figure 6: Root Complex Topology 146
Figure 7: USB Port Configuration 173

List of Tables

Table 1: Arithmetic and Logical Operators ... 22
Table 2: Functions ... 22
Table 3: Operator Precedence and Associativity ... 23
Table 4: Definitions ... 23
Table 5: Processor revision conventions ... 28
Table 6: Compute Unit Definitions ... 35
Table 7: SMM Initial State ... 45
Table 8: SMM Save State ... 45
Table 9: Serial VID Interface (SVI) Port Mapping to SOC Power Planes ... 55
Table 10: Software P-state Naming ... 59
Table 11: Core PMC mapping to PERF_CTL[5:0] ... 75
Table 12: ONION Link Definitions ... 80
Table 13: DCT Definitions ... 81
Table 14: DDR3 UDIMM Maximum Frequency Support FP4/FT4 package ... 83
Table 15: DDR3 SODIMM Maximum Frequency Support FP4/FT4 package ... 83
Table 16: DDR4 UDIMM Maximum Frequency Support ... 83
Table 17: DDR4 SODIMM Maximum Frequency Support FP4 package ... 84
Table 18: DDR4 SODIMM Maximum Frequency Support FT4 package ... 84
Table 19: DDR Population Support Balanced Tee Route ... 84
Table 20: DDR Population Support Daisy Chain or Unbalanced Tee Route ... 84
Table 21: Package Pin Mapping DDR3/DDR4 ... 85
Table 22: DDR Chip to Pad Mapping (DDR3 Mode) ... 86
Table 23: DDR Chip to Pad Mapping (DDR4 Mode) ... 88
Table 24: DCT Definitions ... 90
Table 25: Upstream Mailbox 1 Messages ... 90
Table 26: BIOS Recommendations for DDR3 SODIMM Address and Command Bus ... 95
Table 27: BIOS Recommendations for DDR3 UDIMM Address and Command Bus Configuration ... 98
Table 28: BIOS Recommendations for DDR4 UDIMM Address and Command Bus Configuration ... 101
Table 29: BIOS Recommendations for DDR4 SODIMM Address and Command Bus Configuration ... 103
Table 30: BIOS Recommendations for DDR3 SODIMM Data Bus Configuration ... 105
Table 31: BIOS Recommendations for DDR3 UDIMM Data Bus Configuration ... 108
Table 32: BIOS Recommendations for DDR4 UDIMM Data Bus Configuration ... 111
Table 33: BIOS Recommendations for DDR4 SODIMM Data Bus Configuration ... 113
Table 34: BIOS Recommendations for DDR FIFO RdPtrInitVal ... 115
Table 35: Phy Predriver Codes for DDR3 Data/DQS ... 116
Table 36: Phy Predriver Codes for DDR3 Cmd/Addr ... 116
Table 37: Phy Predriver Codes for DDR3 CLK ... 117
Table 38: Phy Predriver Codes for DDR4 Data/DQS ... 117
Table 39: Phy Predriver Codes for DDR4 Cmd/Addr ... 118
Table 40: Phy Predriver Codes for DDR4 CLK ... 118
Table 41: DDR3 DIMM ODT Pattern ... 121
Table 42: DDR4 DIMM ODT Pattern ... 122

Table 43: DCT Training Specific Register Values 122
Table 44. Configurable Data Pattern Example with 1 Address Target 129
Table 45. Configurable Data Pattern Circular Shift Example with 1 Address Target 130
Table 46. Data Pattern Override Example with 1 Address Target 130
Table 47. Command Generation and Data Comparison 132
Table 48. Command Generation and BubbleCnt Programming 133
Table 49. Recommended Interleave Configurations 134
Table 50. DDR3 Swapped Normalized Address Lines for CS Interleaving 134
Table 51. DDR4 Swapped Normalized Address Lines for CS Interleaving 136
Table 52. Example storage region configuration 140
Table 53: Recommended Interrupt Routing and Swizzling Configuration 147
Table 54: Supported General Purpose (GPP) Link Configurations 148
Table 55: Recommended Legacy Frame Buffer Configurations 156
Table 56: Non-Legacy Mode (Default for Windows® 8.1 and Windows® 10) Carve Out Table 156
Table 57: Frame Buffer Configuration Examples 156
Table 58: MCA Register Cross-Reference Table 159
Table 59: Overwrite Priorities for All Banks 162
Table 60: Error Code Types 163
Table 61: Error Codes: Transaction Type (TT) 163
Table 62: Error Codes: Cache Level (LL) 163
Table 63: Error Codes: Memory Transaction Type (RRRR) 163
Table 64: Error Codes: Participation Processor (PP) 164
Table 65: Error Codes: Memory or IO (II) 164
Table 66: Error Codes: Internal Error Type (UU) 164
Table 67: Error Scope Hierarchy 166
Table 68: Registers Commonly Used for Diagnosis 169
Table 69: USB Port Mapping 173
Table 70: USB Port to EHCI1xB4[PortNumber] Mapping 175
Table 71: SATA PHY Power Saving States 179
Table 72: LPC_SMI Pins 180
Table 73: ASF Remote Control Commands 182
Table 74: Address Mapping Under APB Slave 184
Table 75: Terminology in Register Descriptions 188
Table 76: Reset values for D0F0x64_x3[4:0] 198
Table 77: Mapping for PIF registers 217
Table 78: Mapping for wrapper registers 218
Table 79: Valid Values for D0F2xFC_x20_L1i[3:0] 258
Table 80: Register Mapping for D2F[5:1]x00 288
Table 81: Link controller state encodings 311
Table 82: Register Mapping for D9F2xF4_x1[3:0] 331
Table 83: Register Mapping for D9F2xF4_x2[9:8] 332
Table 84: Reserved field mappings for D9F2xF4_x2[9:8] 333
Table 85: Register Mapping for D9F2xF4_x2[B:A] 333
Table 86: Reserved field mappings for D9F2xF4_x2[B:A] 333

Table 87: Register Mapping for D9F2xF4_x2[D:C] ... 333
Table 88: Reserved field mappings for D9F2xF4_x2[D:C] ... 333
Table 89: Register Mapping for D18F0x[5C:40] ... 336
Table 90: Register Mapping for D18F0x[E4,C4,A4,84] ... 339
Table 91: Register Mapping for D18F0x[EC,CC,AC,8C] ... 340
Table 92: Register Mapping for D18F0x[F0,D0,B0,90] ... 340
Table 93: Link Buffer Definitions ... 341
Table 94: Register Mapping for D18F0x[F4,D4,B4,94] ... 343
Table 95: Register Mapping for D18F0x[F8,D8,B8,98] ... 343
Table 96: Register Mapping for D18F0x[11C,118,114,110] ... 344
Table 97: Register Mapping for D18F0x[18C:170] ... 345
Table 98: Onion Definitions ... 345
Table 99: Register Mapping for D18F0x20[4:0] ... 346
Table 100: Register Mapping for D18F1x[17C:140,7C:40] ... 348
Table 101: Register Mapping for D18F1x[7:4][8,0] ... 349
Table 102: Register Mapping for D18F1x1[7:4][8,0] ... 349
Table 103: Register Mapping for D18F1x[7:4][C,4] ... 350
Table 104: Register Mapping for D18F1x1[7:4][C,4] ... 350
Table 105: Register Mapping for D18F1x[2CC:2A0,1CC:180,BC:80] ... 350
Table 106: Register Mapping for D18F1x[2B:1A,B:8][8,0] ... 351
Table 107: Register Mapping for D18F1x[2B:1A,B:8][C,4] ... 352
Table 108: Register Mapping for D18F1x[2CC:2C0,1CC:1C0,19C:180] ... 353
Table 109: Register Mapping for D18F1x[1F:1E,D:C][8,0] ... 354
Table 110: Register Mapping for D18F1x[1F:1E,D:C][C,4] ... 355
Table 111: Register Mapping for D18F1x[1DC:1D0,EC:E0] ... 356
Table 112: Register Mapping for D18F1x2[1C:00] ... 359
Table 113: Register Mapping for D18F1x2[1,0][8,0] ... 359
Table 114: Register Mapping for D18F1x2[1,0][C,4] ... 360
Table 115: DIMM, Chip Select, and Register Mapping ... 361
Table 116: DDR3 DRAM Address Mapping ... 366
Table 117: DDR4 DRAM Address Mapping ... 366
Table 118: Valid Values for Memory Clock Frequency Value Definition ... 372
Table 119: Index Mapping for D18F2x9C_x00[F,3:0]0_0009_dct[0] ... 374
Table 120: Index Mapping for D18F2x9C_x00[F,3:0]0_0013_dct[0] ... 374
Table 121: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_0014_dct[0] ... 375
Table 122: Index Mapping for D18F2x9C_x00[F,3:0]0_001A_dct[0] ... 375
Table 123: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]028_dct[0] ... 376
Table 124: Index Mapping for D18F2x9C_x00[F,3:0]0_[F,2:0][8,3:0]2E_dct[0] ... 376
Table 125: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] ... 376
Table 126: Index Mapping for D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0] ... 377
Table 127: Index Mapping for D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0] ... 378
Table 128: Index Mapping for D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0] ... 378
Table 129: Index Mapping for D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0] ... 379
Table 130: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_006D_dct[0] ... 379

Table 131: Index Mapping for D18F2x9C_x00[F,3:0]0_0077_dct[0] 380
Table 132: Index Mapping for D18F2x9C_x00[F,3:0]0_0078_dct[0] 380
Table 133: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]081_dct[0] 380
Table 134: Index Mapping for D18F2x9C_x00[F,8:0]1_0000_dct[0] 381
Table 135: Index Mapping for D18F2x9C_x00[F,8:0]1_000E_dct[0] 381
Table 136: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0010_dct[0] 382
Table 137: Index Mapping for D18F2x9C_x00[F,8:0]1_0013_dct[0] 382
Table 138: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0014_dct[0] 383
Table 139: Index Mapping for D18F2x9C_x00[F,8:0]1_0015_dct[0] 383
Table 140: Index Mapping for D18F2x9C_x00[F,8:0]1_0016_dct[0] 384
Table 141: Index Mapping for D18F2x9C_x00[F,8:0]1_001A_dct[0] 385
Table 142: Index Mapping for D18F2x9C_x00[F,8:0]1_001C_dct[0] 385
Table 143: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0] 385
Table 144: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0029_dct[0] 386
Table 145: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_002A_dct[0] 386
Table 146: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_002B_dct[0] 387
Table 147: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_002C_dct[0] 387
Table 148: Index Mapping for D18F2x9C_x00[F,8:0]1_0[8,3:0]2E_dct[0] 388
Table 149: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] 388
Table 150: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]043_dct[0] 389
Table 151: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]044_dct[0] 390
Table 152: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]045_dct[0] 390
Table 153: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]046_dct[0] 391
Table 154: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]047_dct[0] 391
Table 155: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0] 392
Table 156: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] 392
Table 157: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]04E_dct[0] 393
Table 158: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]050_dct[0] 394
Table 159: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]051_dct[0] 394
Table 160: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0] 395
Table 161: Index Mapping for D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0] 395
Table 162: Index Mapping for D18F2x9C_x00[F,8:0]1_0[F,2:0]78_dct[0] 396
Table 163: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0] 396
Table 164: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]81_dct[0] 397
Table 165: Index Mapping for PllMultDiv Value Definition 397
Table 166: Index Mapping for D18F2x9C_x0[3,1:0]02_0080_dct[0] 401
Table 167: Register Mapping for D18F2x1[7C:40]_dct[0] 415
Table 168: Field Mapping for D18F2x1[7C:40]_dct[0] 416
Table 169: BIOS Recommendations for D18F2x1B[4:0] 416
Table 170: Field Mapping for D18F2x1B8_dct[0] 420
Table 171: Field Mapping for D18F2x1BC_dct[0] 421
Table 172: Register Mapping for D18F2x1F[C:0]_dct[0] 422
Table 173: Field Mapping for D18F2x[234:230]_dct[0] 431
Table 174: Field Mapping for D18F2x[23C:238]_dct[0] 431

Table 175: Register Mapping for D18F2x25[8,4]_dct[0] ... 436
Table 176: Register Mapping for D18F2x2[B4,B0,AC,A8]_dct[0] ... 442
Table 177: Field Mapping for D18F2x2[B4,B0,AC,A8]_dct[0] ... 442
Table 178: Field Mappings for D18F2x2[C0,BC]_dct[0] ... 443
Table 179: Field Mappings for D18F2x2[F4:E8]_dct[0]_mp[1:0] ... 445
Table 180: MC0 Error Descriptions ... 453
Table 181: MC0 Error Signatures ... 455
Table 182: MC4 Error Descriptions ... 455
Table 183: MC4 Error Signatures, Part 1 ... 457
Table 184: MC4 Error Signatures, Part 2 ... 458
Table 185: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for All Other Errors ... 459
Table 186: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Protocol Errors ... 459
Table 187: Valid Values for ProtocolErrorType ... 459
Table 188: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for NB Array Errors ... 460
Table 189: Valid Values for ArrayErrorType ... 460
Table 190: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Watchdog Timer Errors ... 461
Table 191: Buffer Definitions ... 464
Table 192: SMAF Action Definition ... 467
Table 193: Register Mapping for D18F3x1[54,50,4C,48] ... 481
Table 194: Register Mapping for D18F4x2[D4:D0] ... 502
Table 195: D18F5x80[Enabled, DualCore] Definition ... 504
Table 196: Register Mapping for D18F5x16[C:0] ... 508
Table 197: NB P-state Definitions ... 508
Table 198: Register Mapping for D18F5x1[FC:C0] ... 513
Table 199: Register Mapping for D18F5x21[8:4] ... 513
Table 200: Register Mapping for GMMx224[C:4] ... 516
Table 201: Register Mapping for IOMMUx[78,70,68,60] ... 528
Table 202: Register Mapping for IOMMUx[7C,74,6C,64] ... 528
Table 203: Register Mapping for IOMMUx[130,128,120,118,110,108,100] ... 530
Table 204: Register Mapping for IOMMUx[134,12C,124,11C,114,10C,104] ... 531
Table 205: Register Mapping for IOMMUx[248,230,218,200] ... 533
Table 206: Register Mapping for IOMMUx[24C,234,21C,204] ... 534
Table 207: Register Mapping for IOMMUx[250,238,220,208] ... 534
Table 208: Register Mapping for IOMMUx[254,23C,224,20C] ... 535
Table 209: Register Mapping for IOMMUx[258,240,228,210] ... 535
Table 210: Register Mapping for IOMMUx[25C,244,22C,214] ... 535
Table 211: Register Mapping for IOMMUx4[1,0][3:0]00 ... 543
Table 212: Register Mapping for IOMMUx4[1,0][3:0]04 ... 543
Table 213: Register Mapping for IOMMUx4[1,0][3:0]08 ... 543
Table 214: Register Mapping for IOMMUx4[1,0][3:0]10 ... 544
Table 215: Register Mapping for IOMMUx4[1,0][3:0]14 ... 544
Table 216: Register Mapping for IOMMUx4[1,0][3:0]18 ... 545
Table 217: Register Mapping for IOMMUx4[1,0][3:0]1C ... 545

Table 218: Register Mapping for IOMMUx4[1,0][3:0]20 ... 545
Table 219: Register Mapping for IOMMUx4[1,0][3:0]24 ... 546
Table 220: Register Mapping for IOMMUx4[1,0][3:0]28 ... 546
Table 221: Register Mapping for IOMMUx4[1,0][3:0]2C ... 547
Table 222: Register Mapping for APIC[170:100] ... 550
Table 223: Register Mapping for APIC[1F0:180] ... 550
Table 224: Register Mapping for APIC[270:200] ... 551
Table 225: ICR valid combinations ... 552
Table 226: Register Mapping for APIC3[60:50] ... 554
Table 227: Div[3,1:0] Value Table ... 556
Table 228: Register Mapping for APIC[4F0:480] ... 557
Table 229: Register Mapping for APIC[530:500] ... 558
Table 230: Reset Mapping for CPUID Fn8000_0000_E[D,C,B]X ... 559
Table 231: CPUID Fn8000_0000_E[B,C,D]X Value ... 567
Table 232: Valid Values for CPUID Fn8000_000[4:2]_E[D,C,B,A]X ... 570
Table 233: ECX mapping to Cache Type for CPUID Fn8000_001D_E[D,C,B,A]X ... 581
Table 234: Register Mapping for MSR0000_020[E,C,A,8,6,4,2,0] ... 591
Table 235: Valid Values for Memory Type Definition ... 592
Table 236: Register Mapping for MSR0000_020[F,D,B,9,7,5,3,1] ... 592
Table 237: Register Mapping for MSR0000_02[6F:68,59:58,50] ... 593
Table 238: Field Mapping for MSR0000_02[6F:68,59:58,50] ... 593
Table 239: MC0 Error Descriptions ... 598
Table 240: MC0 Error Signatures ... 599
Table 241: MC0 Address Register ... 599
Table 242: MC1 Error Descriptions ... 603
Table 243: MC1 Error Signatures ... 605
Table 244: MC1 Address Register ... 606
Table 245: MBE, SBU, and SBC Definitions ... 611
Table 246: MC2 Error Descriptions ... 611
Table 247: MC2 Error Signatures ... 613
Table 248: MC2 Address Register ... 614
Table 249: MC4 Error Descriptions ... 618
Table 250: MC4 Error Signatures, Part 1 ... 619
Table 251: MC4 Error Signatures, Part 2 ... 619
Table 252: MC5 Error Descriptions ... 622
Table 253: MC5 Error Signatures ... 623
Table 254: MC5 Address Register ... 624
Table 255: MC6 Error Descriptions ... 626
Table 256: MC6 Error Signatures ... 627
Table 257: Register Mapping for MSRC001_00[03:00] ... 635
Table 258: Register Mapping for MSRC001_00[07:04] ... 635
Table 259: Register Mapping for MSRC001_00[35:30] ... 640
Table 260: BIOS Recommendations for MSRC001_00[35:30] ... 640
Table 261: Register Mapping for MSRC001_00[53:50] ... 644

Table 262: Register Mapping for MSRC001_00[6B:64] 648
Table 263: P-state Definitions 648
Table 264: Register Mapping for MSRC001_020[A,8,6,4,2,0] 656
Table 265: Register Mapping for MSRC001_020[B,9,7,5,3,1] 658
Table 266: Register Mapping for MSRC001_024[6,4,2,0] 658
Table 267: Register Mapping for MSRC001_024[7,5,3,1] 659
Table 268: Register Mapping for MSRC001_101[B:9] 664
Table 269: Field Mapping for MSRC001_101[B:9] 664
Table 270: Register Mapping for PMCx0D[F:C] 696
Table 271: SATA Controller Subclass Code and ProgramIF Settings 732
Table 272: IDE Compatibility Mode and Native Mode Address Mapping 747
Table 273: Reset mapping for D12F0x8C 782
Table 274: Reset mapping for EHCI1x04 788
Table 275: USB Legacy Registers and IO Ports 829
Table 276: Register Mapping for SDHC0x6[C:0:step4] 849
Table 277: Field Mapping for SDHC0x6[C:0:step4] 850
Table 278: Reset Mapping for GPIOx[0F8:000:step4] 936
Table 279: Debounce Timer Definition 937
Table 280: Reset Mapping for GPIOx[1FC:100:step4] 939
Table 281: Reset Mapping for GPIOx[2DC:200:step4] 941
Table 282: BIOS Recommendations for KbRstEn 992
Table 283: BIOS Recommendations for UsbPhyS5PwrDwnEnable 1000
Table 284: Register Mapping for DMA[1:0]x0[5C:40:Step4] 1052
Table 285: Address Space for I2C controller 1064
Table 286: States for I2C[3:0]x00[IcSlaveDisable] and I2C[3:0]x00[MasterMode] 1064

Revision History

BKDG Revision 3.00 Changes, Jun 8, 2016, PUB release

1 Overview

This document defines AMD Family 15h Models 70h-7Fh Processors, henceforth referred to as the processor.

- The processor overview is located at 2.1 [Processor Overview].
- The processor is distinguished by the combined ExtFamily and BaseFamily fields of the CPUID instruction (see CPUID Fn8000_0001_EAX in 3.19 [CPUID Instruction Registers]).

1.1 Intended Audience

This document provides the processor behavioral definition and associated design notes. It is intended for platform designers and for programmers involved in the development of low-level BIOS (basic input/output system) functions, drivers, and operating system kernel modules. It assumes prior experience in personal computer platform design, microprocessor programming, and legacy x86 and AMD64 microprocessor architecture. The reader should also have familiarity with various platform technologies, such as DDR DRAM.

1.2 Reference Documents

- Advanced Configuration and Power Interface (ACPI) Specification. www.acpi.info.
- *AMD64 Architecture Programmer's Manual Volume 1: Application Programming*, order #24592.
- *AMD64 Architecture Programmer's Manual Volume 2: System Programming*, order #24593.
- *AMD64 Architecture Programmer's Manual Volume 3: Instruction-Set Reference*, order #24594.
- *AMD64 Architecture Programmer's Manual Volume 4: 128-Bit and 256-Bit Media Instructions*, order #26568.
- *AMD64 Architecture Programmer's Manual Volume 5: 64-Bit Media and x87 Floating-Point Instructions*, order #26569.
- *AMD I/O Virtualization Technology™ (IOMMU) Specification*, order #48882.
- *Software Optimization Guide for AMD Family 15h Processors*, order #47414.
- *Revision Guide for AMD Family 15h Models 70h-7Fh Processors*, order #55370.
- JEDEC standards. www.jedec.org.
- PCI local bus specification. (www.pcisig.org).
- PCI Express® specification. (www.pcisig.org).
- Universal Serial Bus Specification (http://www.usb.org)
- Serial ATA Specification (http://www.sata-io.org)
- AT Attachment with Packet Interface (http://www.t13.org)
- SD Host Controller Standard Specification (https://www.sdcard.org)
- Alert Standard Format Specification (http://dmtf.org/standards/asf)

1.3 Conventions

1.3.1 Numbering

- **Binary numbers**. Binary numbers are indicated by appending a “b” at the end (e.g., 0110b).
- **Decimal numbers**. Unless specified otherwise, all numbers are decimal. This rule does not apply to the register mnemonics described in 3.1 [Register Descriptions and Mnemonics]; register mnemonics all utilize hexadecimal numbering.
- **Hexadecimal numbers**. hexadecimal numbers are indicated by appending an “h” to the end (e.g., 45f8h).
- **Underscores in numbers**. Underscores are used to break up numbers to make them more readable. They do not imply any operation (e.g., 0110_1100b).

1.3.2 Arithmetic And Logical Operators

In this document, formulas generally follow Verilog conventions for logic equations.

Table 1: Arithmetic and Logical Operators

Operator	Definition
{}	Concatenation. Curly brackets are used to indicate a group of bits that are concatenated together. Each set of bits is separated by a comma (e.g., {Addr[3:2], Xlate[3:0]} represents a 6-bit value; the two MSBs are Addr[3:2] and the four LSBs are Xlate[3:0]).
\|	Bitwise OR (e.g., 01b \| 10b == 11b).
\|\|	Logical OR (e.g., 01b \|\| 10b == 1b); treats multibit operand as 1 if >=1 and produces a 1-bit result.
&	Bitwise AND (e.g., 01b & 10b == 00b).
&&	Logical AND (e.g., 01b && 10b == 1b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
^	Bitwise exclusive-OR (e.g., (01b ^ 10b == 11b). Sometimes used as "raised to the power of" as well, as indicated by the context in which it is used (e.g., 2^2 == 4).
~	Bitwise NOT. (also known as one's complement) (e.g., ~10b == 01b).
!	Logical NOT (e.g., !10b == 0b); treats multibit operand as 1 if >=1 and produces a 1-bit result.
<, <=, >, >=, ==, !=	Relational. Less than, Less than or equal, greater, greater than or equal, equal, and not equal.
+, -, *, /, %	Arithmetic. Addition, subtraction, multiplication, division, and modulus.
<<	Bitwise left shift. Shift left first operand by the number of bits specified by the 2nd operand (e.g., 01b << 01b == 10b).
>>	Bitwise right shift. Shift right first operand by the number of bits specified by the 2nd operand (e.g., 10b >> 01b == 01b).
?:	Ternary conditional (e.g., *condition* ? *value if true* : *value if false)*. Equivalent to IF *condition* THEN *value if true* ELSE *value if false*.

Table 2: Functions

Function	Definition
ABS	ABS(integer-expression): Remove sign from signed value.
FLOOR	FLOOR(integer-expression): Rounds real number down to nearest integer.
CEIL	CEIL(real-expression): Rounds real number up to nearest integer.
MIN	MIN(integer-expression-list): Picks minimum integer or real value of comma separated list.
MAX	MAX(integer-expression-list): Picks maximum integer or real value of comma separated list.
COUNT	COUNT(integer-expression): Returns the number of binary 1's in the integer.

Table 2: Functions (Continued)

Function	Definition
ROUND	ROUND(real-expression): Rounds to the nearest integer; halfway rounds away from zero.
UNIT	UNIT(fieldName UnitOfMeasure): Input operand is a register field name that defines all values with the same unit of measure. Returns the value expressed in the unit of measure for the current value of the register field.
POW	POW(base, exponent): POW(x,y) returns the value x to the power of y.

1.3.3 Operator Precedence and Associativity

This document follows C operator precedence and associativity. The following table lists operator precedence (highest to lowest). Their associativity indicates in what order operators of equal precedence in an expression are applied. Parentheses are also used to group sub-expressions to force a different precedence; such parenthetical expressions can be nested and are evaluated from inner to outer (e.g., "X = A | ~B & C" is the same as "X = A | ((~B) & C)").

Table 3: Operator Precedence and Associativity

Operator	Description	Associativity
!, ~	Logical negation/bitwise complement	right-to-left
*, /, %	Multiplication/division/modulus	left-to-right
+, -	Addition/subtraction	left-to-right
<<, >>	Bitwise shift left, Bitwise shift right	left-to-right
< , <=, >, >=, ==, !=	Relational operators	left-to-right
&	Bitwise AND	left-to-right
^	Bitwise exclusive OR	left-to-right
\|	Bitwise inclusive OR	left-to-right
&&	Logical AND	left-to-right
\|\|	Logical OR	left-to-right
?:	Ternary conditional	right-to-left

1.4 Definitions

Table 4: Definitions

Term	Definition
AP	Application processor. See 2.3 [Processor Initialization].
BAPM	Bidirectional Application Power Management. See 2.5.8.3 [Bidirectional Application Power Management (BAPM)].
Battery-Power	The system is running from a battery power source or otherwise undocked from a continuous power supply. Setting using this definition may be required to change during runtime.
BCS	Base configuration space. See 2.7 [Configuration Space].

Table 4: Definitions (Continued)

Term	Definition
BERT	Bit error rate tester. A piece of test equipment that generates arbitrary test patterns and checks that a device under test returns them without errors.
BIST	Built-in self-test. Hardware within the processor that generates test patterns and verifies that they are stored correctly (in the case of memories) or received without error (in the case of links).
Boot VID	Boot voltage ID. This is the VDD and VDDNB voltage level that the processor requests from the external voltage regulator during the initial phase of the cold boot sequence. See 2.5.1.2 [Internal VID Registers and Encodings].
BCD	Binary coded decimal number format.
BSC	Boot strap core. Core 0 of the BSP. Specified by MSR0000_001B[BSC].
BSP	Boot strap processor. See 2.3 [Processor Initialization].
CAR	Use of the L2 cache as RAM during boot. See 2.3.3 [Using L2 Cache as General Storage During Boot].
C-states	These are ACPI-defined core power states. C0 is operational. All other C-states are low-power states in which the processor is not executing code. See 2.5.2.2 [Core C-states].
Canonical address	An address in which the state of the most-significant implemented bit is duplicated in all the remaining higher-order bits, up to bit 63.
Channel	See DRAM channel.
CMP	Chip multi-processing. Refers to processors that include multiple cores. See 2.1 [Processor Overview].
COF	Current operating frequency of a given clock domain. See 2.5.2 [CPU Core Power Management].
Cold reset	PWROK is deasserted and RESET_L is asserted. See 2.3 [Processor Initialization].
Compute Unit	Two Cores that share IC, DE, FP and L2 resources. See 2.1 [Processor Overview].
Core	The instruction execution unit of the processor. See 2.1 [Processor Overview].
CPB	Core performance boost. See 2.5.8.1 [Core Performance Boost (CPB)].
CpuCoreNum	Specifies the core number. See 2.4.4 [Processor Cores and Downcoring].
CPUID function X	Refers to the CPUID instruction when EAX is preloaded with X. See 3.19 [CPUID Instruction Registers].
CS	Chip select. See D18F2x[5C:40]_dct[0] [DRAM CS Base Address].
DCT	DRAM controller. See 2.9 [DRAM Controllers (DCTs)].
DCQ	DRAM controller queue.
DDR3	DDR3 memory technology. See 2.9 [DRAM Controllers (DCTs)].
DID	Divisor identifier. Specifies the post-PLL divisor used to reduce the COF. See 2.5.2 [CPU Core Power Management].
Doubleword	A 32-bit value.
Downcoring	Removal of cores. See 2.4.4 [Processor Cores and Downcoring].
DRAM channel	The part of the DRAM interface that connects to a DIMM. See 2.9 [DRAM Controllers (DCTs)].

Table 4: Definitions (Continued)

Term	Definition
Dual-Plane	Refers to a processor or system board where VDD and VDDNB are separate and may operate at independent voltage levels. Refer to 2.5.1 [Processor Power Planes And Voltage Control].
DW	Doubleword. A 32-bit value.
ECS	Extended configuration space. See 2.7 [Configuration Space].
EDS	Electrical data sheet. See 1.2 [Reference Documents].
EFLAGS	See AMD64 Architecture Programmer's Manual Volume 2: System Programming, #24593, section 3.1.6 RFLAGS register, as the legacy EFLAGS register is identical to the low 32 bits of this register.
FCH	The integrated platform device that contains the IO subsystem and the bridge to system BIOS. See 2.17 [FCH].
FDS	Functional data sheet; there is one FDS for each package type.
FID	Frequency identifier. Specifies the PLL frequency multiplier for a given clock domain. See 2.5.2 [CPU Core Power Management].
FreeRunSampleTimer	An internal free running timer used by many power management features. The timer increments at the rate specified by D18F4x110[CSampleTimer].
GB	Gbyte or Gigabyte; 1,073,741,824 bytes.
#GP	A general-protection exception.
#GP(0)	Notation indicating a general-protection exception (#GP) with error code of 0.
GpuEnabled	GpuEnabled = (D1F0x00!=FFFF_FFFFh).
GT/s	Giga-transfers per second.
HCD	Host Controller Driver. A software component.
HTC	Hardware thermal control. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
HTC-active state	Hardware-controlled lower-power, lower-performance state used to reduce temperature. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
IBS	Instruction based sampling. See 2.6.2 [Instruction Based Sampling (IBS)].
IFCM	Isochronous flow-control mode, as defined in the link specification.
ILM	Internal loopback mode. Mode in which the link receive lanes are connected directly to the transmit lanes of the same link for testing and characterization. See D18F0x[18C:170] [Link Extended Control].
IO configuration	Access to configuration space through IO ports CF8h and CFCh. See 2.7 [Configuration Space].
IORR	IO range register. See MSRC001_00[18,16] [IO Range Base (IORR_BASE[1:0])].
IOMMU	IO Memory Management Unit. Also known as AMD Virtualization™ Technology.
KB	Kbyte or Kilobyte; 1024 bytes.
L1 cache	The level 1 caches (instruction cache and the data cache) and the level 2 caches. See 2.1 [Processor Overview].
L2 cache	
Linear (virtual) address	The address generated by a core after the segment is applied.
Link	Generic term that refers to a refer to PCIe® link.
LINT	Local interrupt.

Table 4: Definitions (Continued)

Term	Definition
Logical address	The address generated by a core before the segment is applied.
LVT	Local vector table. A collection of APIC registers that define interrupts for local events (e.g., APIC[530:500] [Extended Interrupt [3:0] Local Vector Table]).
Master abort	This is a PCI-defined term that is applied to transactions on other than PCI buses. It indicates that the transaction is terminated without affecting the intended target; reads return all 1's; writes are discarded; the master abort error code is returned in the response, if applicable; master abort error bits are set if applicable.
MB	Megabyte; 1024 KB.
MCT	Memory controller. See 2.8 [Northbridge (NB)].
MCQ	Memory controller queue. See 2.8 [Northbridge (NB)].
Micro-op	Micro-op. Instructions have variable-length encoding and many perform multiple primitive operations. The processor does not execute these complex instructions directly, but, instead, decodes them internally into simpler fixed-length instructions called macro-ops. Processor schedulers subsequently break down macro-ops into sequences of even simpler instructions called micro-ops, each of which specifies a single primitive operation. See *Software Optimization Guide for AMD Family 15h Processors*.
MEMCLK	Refers to the clock signals, M[B, A][3:0]_CLK, that are driven from the processor to DDR DIMMs.
MMIO	Memory-mapped input-output range. This is physical address space that is mapped to the IO functions such as the IO links or MMIO configuration. The IO link MMIO ranges are specified by D18F1x[2CC:2A0,1CC:180,BC:80] [MMIO Base/Limit].
MMIO configuration	Access to configuration space through memory space. See 2.7 [Configuration Space].
MSR	Model-specific register. The core includes several MSRs for general configuration and control. See 3.20 [MSRs - MSR0000_xxxx] for the beginning of the MSR register definitions.
MTRR	Memory-type range register. The MTRRs specify the type of memory associated with various memory ranges. See MSR0000_00FE, MSR0000_020[F:0], MSR0000_02[6F:68,59:58,50], and MSR0000_02FF.
NB	Northbridge. The transaction routing block of the node. See 2.1 [Processor Overview].
NBC	NBC = (CPUID Fn0000_0001_EBX[LocalApicId[3:0]]==0). Node Base Core. The lowest numbered core in the node.
NBPMC	Performance monitor counter. See 2.6.1.2 [NB Performance Monitor Counters].
NCLK	The main northbridge clock. The NCLK frequency is the NB COF.
Node	See 2.1 [Processor Overview].
Normalized address	Addresses used by DCTs. See 2.8 [Northbridge (NB)].
OW	Octoword. An 128-bit value.
ODM	On-DIMM mirroring. See D18F2x[5C:40]_dct[0][OnDimmMirror].
ODT	On-die termination, which is applied DRAM interface signals.
ODTS	DRAM On-die thermal sensor.
Operational frequency	The frequency at which the processor operates. See 2.5 [Power Management].

Table 4: Definitions (Continued)

Term	Definition
PCIe®	PCI Express®.
PDS	Product data sheet.
Physical address	Addresses used by cores in transactions sent to the NB.
PMC	Performance monitor counter. See 2.6.1.1 [Core Performance Monitor Counters].
PRBS	Pseudo-random bit sequence.
Processor	See 2.1 [Processor Overview].
PSI	Power Status Indicator. See 2.5.1.3.1 [PSIx_L Bit].
P-state	Performance state. See 2.5 [Power Management].
PTE	Page table entry.
QW	Quadword. A 64-bit value.
RAS	Reliability, availability and serviceability (industry term). See 2.16.1 [Machine Check Architecture].
RDQ	Read data queue.
REFCLK	Reference Clock, refers to the clock frequency (100 MHz) or the clock period (10 ns) depending on the context used.
RX	Receiver.
Shutdown	A state in which the affected core waits for either INIT, RESET, or NMI. When shutdown state is entered, a shutdown special cycle is sent on the IO links.
Single-Plane	Refers to a processor or system board where VDD and VDDNB are tied together and operate at the same voltage level. Refer to 2.5.1 [Processor Power Planes And Voltage Control].
Slam	Refers to changing the voltage to a new value in one step (as opposed to stepping).
SMAF	System management action field. This is the code passed from the SMC to the processors in STPCLK assertion messages. The action taken by the processors in response to this message is specified by D18F3x[84:80] [ACPI Power State Control].
SMBus	System management bus. Refers to the protocol on which the serial VID interface (SVI) commands are based. See 2.5.1 [Processor Power Planes And Voltage Control], and 1.2 [Reference Documents].
SMC	System management controller. This is the platform device that communicates system management state information to the processor through an IO link, typically the system IO hub.
SMI	System management interrupt. See 2.4.9.2.1 [SMM Overview].
SMM	System management mode. See 2.4.9.2 [System Management Mode (SMM)].
Speculative event	A performance monitor event counter that counts all occurrences of the event even if the event occurs during speculative code execution.
SVI2	Serial VID 2.0 interface. See 2.5.1.1 [Serial VID Interface].
SVM	Secure virtual machine. See 2.4.10 [Secure Virtual Machine Mode (SVM)].
Sync flood	The propagation of continuous sync packets to all links. This is used to quickly stop the transmission of potentially bad data when there are no other means to do so. See the link specification for additional information.
TCC	Temperature calculation circuit. See 2.10 [Thermal Functions].
T_{ctl}	Processor temperature control value. See 2.10.3 [Temperature-Driven Logic].

Table 4: Definitions (Continued)

Term	Definition
TDC	Thermal design current. See the AMD Infrastructure Roadmap, #41482.
TDP	Thermal design power. A power consumption parameter that is used in conjunction with thermal specifications to design appropriate cooling solutions for the processor. See 2.5.8.2 [Thermal Limiting].
Token	A scheduler entry used in various northbridge queues to track outstanding requests. See D18F3x140 [SRI to XCS Token Count].
TX	Transmitter.
UI	Unit interval. This is the amount of time equal to one half of a clock cycle.
UMI	Unified Media Interface. The link between the processor and the FCH.
VDD	Main power supply to the processor core logic.
VDDNB	Main power supply to the processor NB logic.
VID	Voltage level identifier. See 2.5.1 [Processor Power Planes And Voltage Control].
Virtual CAS	The clock in which CAS is asserted for the burst, N, plus the burst length (in MEMCLKs), minus 1; so the last clock of virtual CAS = N + (BL/2) - 1.
VRM	Voltage regulator module.
W	Word. A 16-bit value.
Warm reset	RESET_L is asserted only (while PWROK stays high). See 2.3 [Processor Initialization].
WDT	Watchdog timer. A timer that detects activity and triggers an error if a specified period of time expires without the activity. For example, see MSRC001_0074 [CPU Watchdog Timer (CpuWdtCfg)] or the NB watchdog timer in D18F3x40 [MCA NB Control].
WDQ	Write data queue.
XBAR	Cross bar; command packet switch. See 2.8 [Northbridge (NB)].

1.5 Changes Between Revisions and Product Variations

Feature support varies by brands and OPNs. To determine the features supported by your processor, contact your customer representative.

1.5.1 Revision Conventions

The processor revision is specified by CPUID Fn0000_0001_EAX [Family, Model, Stepping Identifiers] or CPUID Fn8000_0001_EAX [Family, Model, Stepping Identifiers]. This document uses a revision letter instead of specific model numbers. The following table contains the definitions based on model and stepping used in this document. Where applicable, the processor stepping is indicated after the revision letter. All behavior marked with a revision letter apply to future revisions unless they are superseded by a change in a later revision. See the revision guide for additional information about revision determination. See 1.2 [Reference Documents].

Table 5: Processor revision conventions

Term	Definition
PROC	PROC = {CPUID Fn0000_0001_EAX[ExtFamily], CPUID Fn0000_0001_EAX[ExtModel], CPUID Fn0000_0001_EAX[BaseModel], CPUID Fn0000_0001_EAX[Stepping]}.

1.5.2 Major Changes

This section describes the major changes relative to Family 15h Models 60h-6Fh Processors.

- CPU core changes:
 - 1 Compute Unit containing 2 CPU cores.
 - 1 L2 complex.

- GNB.
 - GPU: GFX 8.1.
 - Universal Video Decoder (UVD) with 10-bit HEVC decode.
 - Video Compression Engine (VCE).
 - Display Control Engine (DCE).
 - Audio Co-processor (ACP).
 - Graphics Memory Controller (GMC).
 - Secure Frame Buffers.

- UNB.
 - Single DRAM Controller with Phy.
 - Memory ECC not supported.

1.5.2.1 Major Changes to Core/NB Performance Counters

Major Changes to Core/NB Performance Counters:

PerfMon Changes: Contains same functionality as Family 15h Models 60h-6Fh.

2 Functional Description

2.1 Processor Overview

The *processor* is defined as follows:
- The processor is a package that contains one node.
- Supports x86-based instruction sets.
- Packages:
 - FP4: Notebook BGA Package.
 - FT4: Notebook Package.
 - AM4: Desktop Package.
 - See CPUID Fn8000_0001_EBX[PkgType].
- Compute Unit
 - 1 Compute Unit (2 cores).
 - 2 MB L2.
 - See 2.4.1 [Compute Unit].
- DRAM:
 - One 64-bit DDR3 or DDR4 memory channel (A).
- Northbridge:
 - One communication packet routing block referred to as the northbridge (NB). The NB routes transactions between the cores, the link, and the DRAM interfaces. It includes the configuration register space for the device.
- Graphics Northbridge:
 - GPU: graphics core with 3 Compute Units and one 1 Render Backend+.

- Number of independent displays: 2.
- Number of display interfaces: 3.
- Full HSA2.0 Enabled, OpenGL ES™, OpenCL®, DirectX® 11 and 12.
 - Erickson Texture Compression (ETC2) for OpenGL 4.3 and OpenGL ES 3.0 support.
- Render Backend Plus (RB+):
 - Improves the color fill rates.
 - Improved 2D graphics workloads.

- Universal Video Decoder.
 - H.264 with support for up to 20 HD streams.
 - H.265 (10-bit HEVC with Dithering support).
 - MPEG2, VC1, WMV9, DivX®, and MPEG-4 video decoding.
 - JPEG (Baseline Profile, up to 8Kx8K).
- Video Compression Engine.
- Display Controller Engine, supporting up to 4K x 2K 60fps resolution.
 - Hybrid Memory Carve Out:
 - Secure memory region for UHD (4K) premium content.
 - Allows content providers to counter piracy with IP protection.
- Audio Co-processor.
- Graphics Memory Controller.
- Secure Virtual Machine (SVM-lite).
 - Non-zero Virtual Machine ID for Multimedia Engines.
 - Allows multimedia engines to access virtual memory within the address space, instead of physical memory.
- Link:
 - PCIe® Gen3 on the GPP links.
 - 8 GPP lanes.
 - See 2.11.3 [Links].

- Integrated FCH:
 - AOAC (Connected/Modern Standby) Support.
 - 4 Super Speed USB3 ports.
 - 8 USB2/USB1.1 ports.
 - 2 UART ports.
 - 4 I^2C ports.
 - 2 SATA Gen3 ports.
 - SD/SDIO v3.0 port.
 - LPC/SPI (100 MHz), eSPI.
 - 2 SMBus Ports.
- Power Management:
 - AVFS (Adaptive Frequency & Voltage Scaling) supported on the x86 Compute Unit.
 - 8 bits of LPML fusing.
- RAS:
 - See 2.16 [RAS Features].

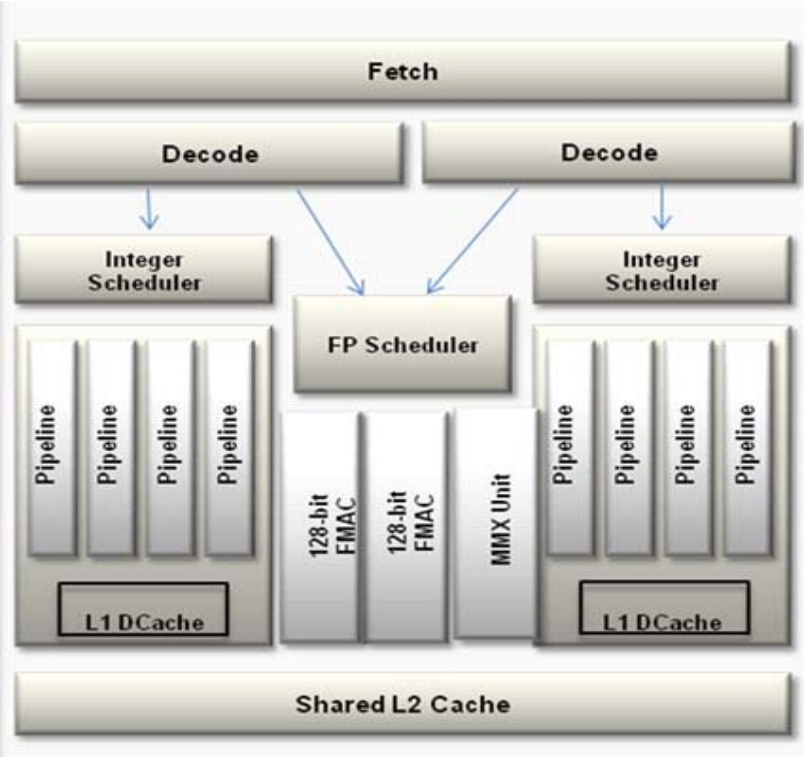


Figure 1: A Compute Unit

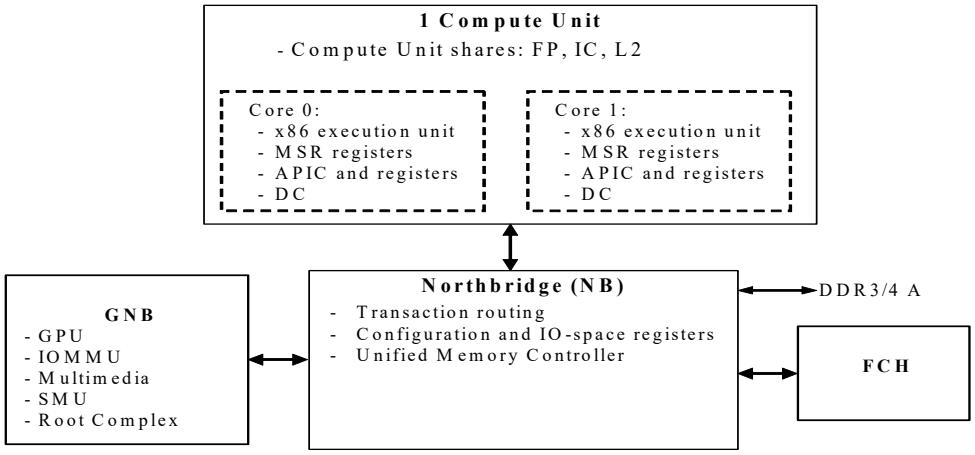


Figure 2: A Processor

2.2 System Overview

2.3 Processor Initialization

This section describes the initialization sequence after a cold reset.

Core 0 of the processor, the bootstrap core (BSC), begins executing code from the reset vector. The remaining cores do not fetch code until their enable bits are set (D18F0x1DC[CpuEn]).

2.3.1 BSC Initialization

The BSC must perform the following tasks as part of boot.

- Store BIST information from the EAX register into an unused processor register.
- D18F0x6C[InitDet] may be used by BIOS to differentiate between INIT and cold/warm reset.
- Determine type of startup using D18F0x6C[ColdRstDet].
 - If this is a warm reset then BIOS may check for valid MCA errors and if present save the status for later use. See 2.16.1.6 [Handling Machine Check Exceptions].
- Enable the cache, program the MTRRs for CAR and initialize CAR. See 2.3.3 [Using L2 Cache as General Storage During Boot].
- Setup the SMU.
- Setup of APIC (2.4.9.1.3 [ApicId Enumeration Requirements]).
- Setup the link configuration.

- Setup the root complex and initialize the IO links.
- If required, reallocate data and flow control buffers of the links (see D18F0x[F0,D0,B0,90] [Link Base Channel Buffer Count] and D18F0x[F4,D4,B4,94] [Link Isochronous Channel Buffer Count]).
- Issue system warm reset.
- Configure the DRAM controllers.
- Configure processor power management. See 2.5 [Power Management].
- Allow other cores to begin fetching instructions by setting D18F0x1DC[CpuEn] in the PCI configuration space of all nodes. See 2.4.4 [Processor Cores and Downcoring].

2.3.2 AP Initialization

All other processor cores other than core 0 begin executing code from the reset vector. They must perform the following tasks as part of boot.

- Store BIST information from the EAX register into an unused processor register.
- D18F0x6C[InitDet] may be used by BIOS to differentiate between INIT and cold/warm reset.
- Determine the history of this reset using the D18F0x6C [Link Initialization Control] [ColdRstDet] bit:
 - If this is a warm reset then BIOS may check for valid MCA errors and if present save the status for use later. See 2.16.1.6 [Handling Machine Check Exceptions].
- Set up the local APIC. See 2.4.9.1.3 [ApicId Enumeration Requirements].
- Configure processor power management. See 2.4 [Core].

2.3.3 Using L2 Cache as General Storage During Boot

Prior to initializing the DRAM controller for system memory, BIOS may use the L2 cache of each core as general storage.

The L2 cache as general storage is described as follows:
- Each Compute Unit has its own L2 cache.
- BIOS manages the mapping of the L2 storage such that cacheable accesses do not cause L2 victims.
- The L2 size, L2 associativity, and L2 line size is determined by reading CPUID Fn8000_0006_ECX[L2Size, L2Assoc, L2LineSize]. L2WayNum is defined to be the number of ways indicated by the L2Assoc code.
 - The L2 cache is viewed as (L2Size/L2LineSize) cache lines of storage, organized as L2WayNum ways, each way being (L2Size/L2WayNum) in size.
 (e.g., L2Assoc=8 so L2WayNum=16 (there are 16 ways). If (L2Size=2MB) then there are 16 blocks of cache, each 2MB/16 in size, or 128KB each.)
 - For each of the following values of L2Size, the following values are defined:
 - L2Size=1 MB: L2Tag=PhysAddr[39:16], L2WayIndex=PhysAddr[15:6].
 - PhysAddr[5:0] addresses the L2LineSize number of bytes of storage associated with the cache line.
 - The L2 cache, when allocating a line at L2WayIndex:
 - Picks an invalid way before picking a valid way.
 - Prioritizes the picking of invalid ways such that way L2WayNum-1 is the highest priority and 0 is the lowest priority.
 - It is recommended that BIOS assume a simpler allocation of L2 cache memory, being L2WayNum size-aligned blocks of memory, each being L2Size/L2WayNum bytes.
 - BIOS can rely on a minimum L2Size of 256 KB and can rely on being able to use a minimum of 14 ways for general storage. See CPUID Fn8000_0006_ECX[L2Size]. See initialization requirements below for MSRC001_1023[L2WayLock, L2FirstLockedWay].

The following memory types are supported:
- WP-IO: BIOS ROM may be assigned the write-protect IO memory type and may be accessed read-only

as data and fetched as instructions.
 - WP-IO accesses, both read and write, do not get evicted to the L2 and therefore do not need to be considered for allocation into the L2.
 - WB-DRAM: General storage may be assigned the write-back DRAM memory type and may be accessed as read-write data, but not accessed by instruction fetch.
 - BIOS initializes an L2LineSize sized and aligned location in the L2 cache, mapped as write-back DRAM, with 1 read to at least 1 byte of the L2LineSize sized and aligned WB-DRAM address. BIOS may store to a line only after it has been allocated by a load.
 - Fills, sent to the disabled memory controller, return undefined data.
 - All of memory space that is not accessed as WP-IO or WB-DRAM space must be marked as UC memory type.

- In order to prevent victimizing L2 data, no more than L2WayNum cache lines accessed as WB-DRAM may have the same L2WayIndex.
 - Software does not need to know which ways the L2WayNum lines are allocated to for any given value of L2WayIndex, only that invalid ways are selected for allocation before valid ways are selected for allocation.
 - Software is not allowed to deallocate a line in the L2 by using CLFLUSH.

Performance monitor event PMCx07F[1], titled “L2 Writebacks to system“, can be used to indicate whether L2 dirty data was lost by being victimized and sent to the disabled memory controller.

The following requirements must be satisfied prior to using the cache as general storage:
- Paging must be disabled.
- MSRC001_0015[INVDWBINVD]=0.
- MSRC001_101C[DisSS]=1.
- MSRC001_1021[DisSpecTlbRld]=1. Disable speculative ITLB reloads.
- MSRC001_101C[DisSpecTlbRld]=1. Disable speculative DTLB reloads.
- MSRC001_101C[DisHwPf]=1.
- MSRC001_102B[CombineCr0Cd]=0. See MSRC001_102B[CombineCr0Cd].
- CLFLUSH, INVD, and WBINVD must not be used during CAR but may be used when tearing down CAR for all compute units on a node.
- The BIOS must not use SSE, or MMX™ instructions, with the exception of the following list: MOVD, MOVQ, MOVDQA, MOVQ2DQ, MOVDQ2Q.
- The BIOS must not enable exceptions, page-faults, and other interrupts.
- BIOS must not use software prefetches.
- UC-DRAM: All of DRAM that is not accessed as WB-DRAM space must be marked as UC memory type.
- If (MSRC001_1023[L2WayLock]==1) then:
 - Only the ways 0 through (MSRC001_1023[L2FirstLockedWay]-1) may be used for general storage.
 - BIOS can rely on MSRC001_1023[L2FirstLockedWay] to have a minimum value of Eh.
- If (MSRC001_1023[L2WayLock]==0) then:
 - Set MSRC001_1023[L2WayLock]=1.
 - Set MSRC001_1023[L2FirstLockedWay]=Fh.

When BIOS has completed using the cache for general storage the following steps are followed:
1. An INVD instruction is executed on each core that used cache as general storage; an INVD is issued when all cores on all nodes have completed using the cache for general storage.
2. If DRAM is initialized and there is data in the cache that needs to get moved to main memory, CLFLUSH or WBINVD may be used instead of INVD, but software must ensure that needed data in main memory is not overwritten.

3. Program the following configuration state (Order is unimportant):
 - MSRC001_0015[INVDWBINVD]=1.
 - MSRC001_101C[DisSS]=0.
 - MSRC001_1021[DisSpecTlbRld]=0.
 - MSRC001_101C[DisSpecTlbRld]=0.
 - MSRC001_101C[DisHwPf]=0.
 - If ((MSRC001_1023[L2WayLock]==1) & (MSRC001_1023[L2FirstLockedWay]==Fh)), program MSRC001_1023[L2WayLock]=0.

2.4 Core

The majority of the behavioral definition of the core is specified in the AMD64 Architecture Programmer's Manual. See 1.2 [Reference Documents].

2.4.1 Compute Unit

A *compute unit* includes 2 cores each having an x86 instruction execution logic and first-level (L1) data cache. The FP unit, second level (L2) general-purpose cache, and first-level instruction cache, are shared between both cores of the compute unit.
There is a set of MSRs and APIC registers associated with each core. Processors that include multiple cores are said to incorporate *chip multi-processing* or CMP. Unless otherwise specified the processor configuration interface hides the Compute Unit implementation and presents software with homogenous cores, each independent of the other.
Software may use D18F5x80[Enabled, DualCore] in order to associate a core with a Compute Unit. This information can be useful because some configuration settings are determined based on active Compute Units and core performance may vary based on resource sharing within a Compute Unit.

Table 6: Compute Unit Definitions

Term	Definition
NumOfCompUnits	The number of Compute Units for which at least 1 core is enabled. NumOfCompUnits = COUNT(D18F5x80[Enabled]).
DualCoreEnabled	Both cores of a compute unit are enabled. DualCoreEnabled = (D18F5x80[DualCore[0]]==1). 0=Core 0 enabled, Core 1 disabled.

2.4.2 Caches and TLBs

Cache and TLBstorage available to a core is reported by:
- CPUID Fn8000_0005_EAX-CPUID Fn8000_0006_EDX.
- CPUID Fn8000_0019_EAX-CPUID Fn8000_0019_E[D,C]X.
- CPUID Fn8000_001D_EAX_x0-CPUID Fn8000_001E_EDX.

Cache and TLB storage available to a core is summarized as follows:
- L1 and L2 Caches:
 - DC: 32 KB, 8-way, write-through, per-core.
 - IC: 96 KB, 2-way, shared between cores of a compute unit.
 - L2: 1 MB or 2MB (Product-specific), 16-way associative, shared between both cores of a compute unit.
- TLBs:
 - D, L1TLB:
 - 4 KB: 32 entries, fully associative.

 - 2 MB: 32 entries, fully associative.
 - 1 GB: 32 entries, fully associative.
- D, L2TLB:
 - None. Full size of unified TLB reported as L2 DTLB.
- I, L1TLB:
 - 4 KB: 48 entries, fully associative.
 - 2 MB, 1 GB: 24 entries, fully associative. 2M and 1G entries share the same L1TLB bank.
- I, L2TLB:
 - 4 KB: 512 entries, 4-way associative. (Same as Fam10h)
 - 2 MB: None. Full size of unified TLB reported as L2 ITLB.
 - 1 GB: None. Full size of unified TLB reported as L2 ITLB.
- Unified TLB:
 - 1024 entries, 8-way associative, any entry can cache:
 - D: 4K, 2M, 4M, or 1G translation.
 - I: 2M, 4M, or 1G translation. Not 4K ITLB translations.
 - Notes: Unified TLB natively stores 4M translations. An entry allocated by one core is not visible to the other core of a compute unit.

2.4.2.1 Registers Shared by Cores in a Compute Unit

Some registers are implemented one instance per Compute Unit instead of per core; these registers are designated as Per-compute-unit. The absence of Per-compute-unit implies the normal per-core instance programming model.
Some Per-compute-unit MSRs are implemented as registers that when read the contents are saved in the L1 data cache and are not coherent between cores; these registers are called SharedNC or “shared non-coherent”.

Programing rules for Per-compute-unit registers:
- Software must ensure that a shared MSR written by one core on a Compute Unit does not cause a problem for software that is running on the other core of the Compute Unit.
- Not SharedNC: A write to a MSR does not have to be written to the other core of the compute unit in order for the other core to see the updated value.
- SharedNC: A write to a SharedNC MSR has to be written to both cores of the compute unit in order for both cores to see the updated value.
 - If software can know that the other core has not read the SharedNC MSR since the last warm reset, then a write is not needed to the SharedNC MSR on the other core.
 - Software may not rely on the other core maintaining the previous value of the SharedNC MSR.
- A read-modify-write of a shared MSR register is not atomic. Software must ensure atomicity between the cores that could simultaneously read-modify-write the shared register.

2.4.3 Virtual Address Space

The processor supports 48 address bits of virtual memory space (256 TB) as indicated by CPUID Fn8000_0008_EAX.

If the memory maps a region as DRAM that is not-populated with real storage behind it, that area of DRAM must be mapped as non-cachable (UC memory type).

2.4.4 Processor Cores and Downcoring

Each node supports downcoring as follows:
- The number of cores supported is specified by D18F5x84[CmpCap].

- The cores of a compute unit may be software downcored by D18F3x190[DisCore] if (DualCoreEnabled==1). If (DualCoreEnabled==0) then the cores of a compute unit may not be software downcored. See 2.4.4.1 [Software Downcoring using D18F3x190[DisCore]].
 - All cores of a compute unit must be downcored if either core needs to be downcored.
 - D18F3x190[DisCore] affects the value of CPUID Fn0000_0001_EBX[LogicalProcessorCount], CPUID Fn0000_0001_EDX[HTT], CPUID Fn8000_0001_ECX[CmpLegacy], CPUID Fn8000_0008_ECX[NC], D18F5x80[Enabled, DualCore]. D18F3x190[DisCore] does not affect the value of D18F5x84[CmpCap].
- An implemented (physical) core that is downcored is not visible to software. Cores that are not downcored are numbered logically in a contiguous manner.
- D18F5x80 [Compute Unit Status 1] reports core topology information to software.
- The number of cores specified in CPUID Fn8000_0008_ECX[NC] must be the same as the number of cores enabled in D18F0x1DC[CpuEn].
- The core number, *CpuCoreNum*, is provided to software running on each core through CPUID Fn0000_0001_EBX[LocalApicId] and APIC20[ApicId]; CpuCoreNum also affects D18F0x1DC[CpuEn]. CpuCoreNum, varies as the lowest integers from 0 to D18F5x84[CmpCap], based on the number of enabled cores (e.g., a 4-core node with 1 core disabled results in cores reporting CpuCoreNum values of 0, 1, and 2 regardless of which core is disabled). The boot core is always the core reporting CpuCoreNum=0.

Some legacy operating systems do not support processors with a non-power-of-2 number of cores. The BIOS is recommended to support a user configurable option to disable cores down to a power-of-2 number of cores for legacy operating system support.

2.4.4.1 Software Downcoring using D18F3x190[DisCore]

Cores may be downcored by D18F3x190[DisCore].

Software is required to use D18F3x190[DisCore] as follows:

- Setting bits corresponding to cores that are not present results in undefined behavior.
- Once a core has been removed by D18F3x190[DisCore]=1, it cannot be added back without a cold reset (e.g., Software may only set DisCore bits, never clear them).
- Software may remove cores only once. If software removes cores by setting D18F3x190[DisCore]=1, then software is not allowed to disable additional cores after the next warm reset.
- The most significant bit N is (the number of cores)-1 at cold reset; the number of cores at cold reset is (CPUID Fn8000_0008_ECX[NC]+1).
- The most significant bit N and the core ID significance of DisCore is not affected by the value of DisCore followed by a warm-reset (e.g., If core 2 is disabled by DisCore[3:0]=0100b followed by a warm reset, then the new core 2 is the old core 3. If the new core 2 needs to then be disabled then DisCore[3:0] = 1100b followed by a warm reset).
- All bits greater than bit N are reserved.
- If D18F3x190[DisCore] is changed, then the following need to be updated:
 - D18F0x60[CpuCnt[4:0]].
 - D18F5x170[NbPstateThreshold].
 - MSRC001_102A[ThrottleNbInterface].

2.4.5 Physical Address Space

The processor supports a 40 bit physical address space, even though the core indicates support for a 48 bit physical address space. See CPUID Fn8000_0008_EAX [Long Mode Address Size Identifiers].

The processor master aborts the following upper-address transactions (to address PhysAddr):

- Link or core requests with non-zero PhysAddr[63:40].

2.4.6 System Address Map

The processor defines a reserved memory address region starting at 0000_00FD_0000_0000h and extending up to 0000_00FF_FFFF_FFFFh. System software must not map memory into this region. Downstream host accesses to the reserved address region results in a page fault. Upstream system device accesses to the reserved address region results in an undefined operation.

2.4.6.1 Memory Access to the Physical Address Space

All memory accesses to the physical address space from a core are sent to its associated northbridge (NB). All memory accesses from a link are routed through the NB. An IO link access to physical address space indicates to the NB the cache attribute (Coherent or Non-coherent, based on bit[0] of the Sized Read and Write commands).

A core access to physical address space has two important attributes that must be determined before issuing the access to the NB: the memory type (e.g., WB, WC, UC; as described in the MTRRs) and the access destination (DRAM or MMIO).

This mechanism is managed by the BIOS and does not require any setup or changes by system software.

2.4.6.1.1 Determining Memory Type

The memory type for a core access is determined by the highest priority of the following ranges that the access falls in: 1==Lowest priority.

1. The memory type as determined by architectural mechanisms.
 - See the APM2 chapter titled “Memory System”, sections “Memory-Type Range Registers” and “Page-Attribute Table Mechanism”.
 - See the APM2 chapter titled “Nested Paging”, section “Combining Memory Types, MTRRs”.
 - See MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)], MSR0000_020[F:0] [Variable-Size MTRRs Base/Mask], MSR0000_02[6F:68,59:58,50] [Fixed-Size MTRRs].
2. TSeg & ASeg SMM mechanism. (see MSRC001_0112 and MSRC001_0113)
3. CR0[CD]: If (CR0[CD]==1) then MemType=CD.
4. MMIO configuration space, APIC space.
 - MMIO APIC space and MMIO config space must not overlap.
 - MemType=UC.
 - See 2.4.9.1.2 [APIC Register Space] and 2.7 [Configuration Space].
5. If (“In SMM Mode” && ~((MSRC001_0113[AValid] && “The address falls within the ASeg region”) || (MSRC001_0113[TValid] && “The address falls within the TSeg region”))) then MemType=CD.

2.4.6.1.2 Determining The Access Destination for Core Accesses

The access destination, DRAM or MMIO, is based on the highest priority of the following ranges that the access falls in: 1==Lowest priority.

1. RdDram/WrDram as determined by MSRC001_001A [Top Of Memory (TOP_MEM)] and MSRC001_001D [Top Of Memory 2 (TOM2)].
2. The IORRs. (see MSRC001_00[18,16] and MSRC001_00[19,17]).
3. The fixed MTRRs. (see MSR0000_02[6F:68,59:58,50] [Fixed-Size MTRRs]).
4. TSeg & ASeg SMM mechanism. (see MSRC001_0112 and MSRC001_0113).
5. MMIO config space, APIC space.
 - MMIO APIC space and MMIO config space must not overlap.

- RdDram=IO, WrDram=IO.
- See 2.4.9.1.2 [APIC Register Space] and 2.7 [Configuration Space].

6. NB address space routing. See 2.8.2.1.1 [DRAM and MMIO Memory Space].

2.4.7 Timers

Each core includes the following timers. These timers do not vary in frequency regardless of the current P-state or C-state.

- MSR0000_0010 [Time Stamp Counter (TSC)]; the TSC increments at the rate specified by the P0 P-state.
 - See 2.5.2.1.1.1 [Software P-state Numbering].
 - See MSRC001_00[6B:64] [P-state [7:0]].
- The APIC timer (APIC380 and APIC390), which increments at the rate of 2xCLKIN; the APIC timer may increment in units of between 1 and 8.

2.4.8 Implicit Conditions for TLB Invalidation

The following family specific conditions cause all TLBs for both cores of the compute unit to be invalidated; except MSR0000_0277 which clears the TLBs for the core that did the MSR write. The architectural conditions that cause TLB invalidation are documented by the APM2 section titled “Translation-Lookaside Buffer (TLB)”; see “Implicit Invalidations”.

- MSR0000_020[F:0] [Variable-Size MTRRs Base/Mask].
- MSR0000_02[6F:68,59:58,50] [Fixed-Size MTRRs].
- MSR0000_0277 [Page Attribute Table (PAT)] (TLBs not cleared for the other core).
- MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)].
- MSRC001_0010 [System Configuration (SYS_CFG)] write.
- MSRC001_00[18,16] [IO Range Base (IORR_BASE[1:0])] write.
- MSRC001_00[19,17] [IO Range Mask (IORR_MASK[1:0])] write.
- MSRC001_001A [Top Of Memory (TOP_MEM)] write.
- MSRC001_001D [Top Of Memory 2 (TOM2)] write.
- MSRC001_1023 [Combined Unit Configuration (CU_CFG)] write.
- MSRC001_102A [Combined Unit Configuration 2 (CU_CFG2)] write.
- MSRC001_102B [Combined Unit Configuration 3 (CU_CFG3)] write.

2.4.9 Interrupts

2.4.9.1 Local APIC

The local APIC contains logic to receive interrupts from a variety of sources and to send interrupts to other local APICs, as well as registers to control its behavior and report status. Interrupts can be received from:

- IO devices including the IO hub (IO APICs).
- Other local APICs (inter-processor interrupts).
- APIC timer.
- Thermal events.
- Performance counters.
- Legacy local interrupts from the IO hub (INTR and NMI).
- APIC internal errors.

The APIC timer, thermal events, performance counters, local interrupts, and internal errors are all considered local interrupt sources, and their routing is controlled by local vector table entries. These entries assign a message type and vector to each interrupt, allow them to be masked, and track the status of the interrupt.

IO and inter-processor interrupts have their message type and vector assigned at the source and are unaltered by the local APIC. They carry a destination field and a mode bit that together determine which local APIC(s) accepts them. The destination mode (DM) bit specifies if the interrupt request packet should be handled in physical or logical destination mode. If the destination field matches the broadcast value specified by D18F0x68[ApicExtBrdCst], then the interrupt is a broadcast interrupt and is accepted by all local APICs regardless of destination mode.

2.4.9.1.1 Detecting and Enabling

APIC is detected and enabled via CPUID Fn0000_0001_EDX[APIC].

The local APIC is enabled via MSR0000_001B[ApicEn]. Reset forces APIC disabled.

2.4.9.1.2 APIC Register Space

MMIO APIC space:

- Memory mapped to a 4 KB range. The memory type of this space is the UC memory type. The base address of this range is specified by {MSR0000_001B[ApicBar[47:12]], 000h}.
- The mnemonic is defined to be APICXX; XX is the byte address offset from the base address.
- MMIO APIC registers in xAPIC mode is defined by the register from APIC20 to APIC[530:500].
- Treated as normal memory space when APIC is disabled, as specified by MSR0000_001B[ApicEn].

If the memory maps a region as DRAM that is not-populated with real storage behind it, that area of DRAM must be mapped as non-cachable (UC memory type).

2.4.9.1.3 ApicId Enumeration Requirements

System hardware and BIOS must ensure that the number of cores per processor (NC) exposed to the operating system by all tables, registers, and instructions across all cores in the processor is identical. See 2.4.11.1 [Multi-Core Support] to derive NC.

Operating systems are expected to use CPUID Fn8000_0008_ECX[ApicIdCoreIdSize], the number of least significant bits in the Initial APIC ID that indicate core ID within a processor, in constructing per-core CPUID masks. (ApicIdCoreIdSize[3:0] determines the maximum number of cores (MNC) that the processor could theoretically support, not the actual number of cores that are actually implemented or enabled on the processor, as indicated by CPUID Fn8000_0008_ECX[NC].) MNC = (2 ^ CPUID Fn8000_0008_ECX[ApicIdCoreIdSize]). BIOS must use the ApicId MNC rule when assigning APIC20[ApicId] values as described below.

ApicId MNC rule: The ApicId of core j on processor i must be enumerated/assigned as:

- ApicId[proc=i, core=j] = (OFFSET_IDX + i) * MNC + j
- Where OFFSET_IDX is an integer offset (0 to N) used to shift up the core ApicId values to allow room for IOAPIC devices.

It is recommended that BIOS use the following APIC ID assignments for the broadest operating system support. Given N = (Number_Of_Processors * MNC) and M = Number_Of_IOAPICs:

- If (N+M) < 16, then assign the local (core) ApicIds first from 0 to N-1, and the IOAPIC IDs from N to N+(M-1). APIC ID 15 is reserved for broadcast when APIC410[ExtApicIdEn]==0.
- If (N+M) >= 16, then assign the IOAPIC IDs first from 0 to M-1, and the local (core) ApicIds from K to K+(N-1), where K is an integer multiple of MNC greater than M-1.

2.4.9.1.4 Physical Destination Mode

The interrupt is only accepted by the local APIC whose APIC20[ApicId] matches the destination field of the interrupt. Physical mode allows up to 255 APICs to be addressed individually.

2.4.9.1.5 Logical Destination Mode

A local APIC accepts interrupts selected by APICD0 [Logical Destination (LDR)] and the destination field of the interrupt using either cluster or flat format as configured by APICE0[Format].

If flat destinations are in use, bits[7:0] of APICD0[Destination] are checked against bits[7:0] of the arriving interrupt's destination field. If any bit position is set in both fields, the local APIC is a valid destination. Flat format allows up to 8 APICs to be addressed individually.

If cluster destinations are in use, bits[7:4] of APICD0[Destination] are checked against bits[7:4] of the arriving interrupt's destination field to identify the cluster. If all of bits[7:4] match, then bits[3:0] of APICD0[Destination] and the interrupt destination are checked for any bit positions that are set in both fields to identify processors within the cluster. If both conditions are met, the local APIC is a valid destination. Cluster format allows 15 clusters of 4 APICs each to be addressed.

2.4.9.1.6 Interrupt Delivery

SMI, NMI, INIT, Startup, and External interrupts are classified as non-vectored interrupts.

When an APIC accepts a non-vectored interrupt, it is handled directly by the processor instead of being queued in the APIC. When an APIC accepts a fixed or lowest-priority interrupt, it sets the bit in APIC[270:200] [Interrupt Request (IRR)] corresponding to the vector in the interrupt. For local interrupt sources, this comes from the vector field in that interrupt's local vector table entry. The corresponding bit in APIC[1F0:180] [Trigger Mode (TMR)] is set if the interrupt is level-triggered and cleared if edge-triggered. If a subsequent interrupt with the same vector arrives when the corresponding bit in APIC[270:200][RequestBits] is already set, the two interrupts are collapsed into one. Vectors 15-0 are reserved.

2.4.9.1.7 Vectored Interrupt Handling

APIC80 [Task Priority (TPR)] and APICA0 [Processor Priority (PPR)] each contain an 8-bit priority divided into a main priority (bits[7:4] and a priority sub-class (bits[3:0]. The task priority is assigned by software to set a threshold priority at which the processor is interrupted.

The processor priority is calculated by comparing the main priority (bits[7:4]) of APIC80[Priority] to bits[7:4] of the 8-bit encoded value of the highest bit set in APIC[170:100] [In-Service (ISR)]. The processor priority is the higher of the two main priorities.

The processor priority is used to determine if any accepted interrupts (indicated by APIC[270:200][RequestBits]) are high enough priority to be serviced by the processor. When the processor is ready to service an interrupt, the highest bit in APIC[270:200][RequestBits] is cleared, and the corresponding bit is set in APIC[170:100][InServiceBits].

When the processor has completed service for an interrupt, it performs a write to APICB0 [End of Interrupt], clearing the highest bit in APIC[170:100][InServiceBits] and causing the next-highest interrupt to be serviced. If the corresponding bit in APIC[1F0:180][TriggerModeBits] is set, a write to APICB0 is performed on all APICs to complete service of the interrupt at the source.

2.4.9.1.8 Interrupt Masking

Interrupt masking is controlled by the APIC410 [Extended APIC Control]. If APIC410[IerEn] is set, APIC[4F0:480] [Interrupt Enable] are used to mask interrupts. Any bit in APIC[4F0:480][InterruptEnableBits] that is clear indicates the corresponding interrupt is masked. A masked interrupt is not serviced and the corresponding bit in APIC[270:200][RequestBits] remains set.

2.4.9.1.9 Spurious Interrupts

In the event that the task priority is set to or above the level of the interrupt to be serviced, the local APIC delivers a spurious interrupt vector to the processor, as specified by APICF0 [Spurious-Interrupt Vector (SVR)]. APIC[170:100] is not changed and no write to APICB0 occurs.

2.4.9.1.10 Spurious Interrupts Caused by Timer Tick Interrupt

A typical interrupt is asserted until it is serviced. An interrupt is deasserted when software clears the interrupt status bit within the interrupt service routine. Timer tick interrupt is an exception, since it is deasserted regardless of whether it is serviced or not.

The processor is not always able to service interrupts immediately (i.e., when interrupts are masked by clearing EFLAGS.IM).

If the processor is not able to service the timer tick interrupt for an extended period of time, the INTR caused by the first timer tick interrupt asserted during that time is delivered to the local APIC in ExtInt mode and latched, and the subsequent timer tick interrupts are lost. The following cases are possible when the processor is ready to service interrupts:

- An ExtInt interrupt is pending, and INTR is asserted. This results in timer tick interrupt servicing. This occurs 50 percent of the time.
- An ExtInt interrupt is pending, and INTR is deasserted. The processor sends the interrupt acknowledge cycle, but when the PIC receives it, INTR is deasserted, and the PIC sends a spurious interrupt vector. This occurs 50 percent of the time.

There is a 50 percent probability of spurious interrupts to the processor.

2.4.9.1.11 Lowest-Priority Interrupt Arbitration

Fixed, remote read, and non-vectored interrupts are accepted by their destination APICs without arbitration.

Delivery of lowest-priority interrupts requires all APICs to arbitrate to determine which one accepts the interrupt. If APICF0[FocusDisable] is clear, then the focus processor for an interrupt always accepts the interrupt. A processor is the focus of an interrupt if it is already servicing that interrupt (corresponding bit in APIC[170:100][InServiceBits] is set) or if it already has a pending request for that interrupt (corresponding bit in APIC[270:200][RequestBits] is set). If APIC410[IerEn] is set the interrupt must also be enabled in APIC[4F0:480][InterruptEnableBits] for a processor to be the focus processor. If there is no focus processor for an interrupt, or focus processor checking is disabled, then each APIC calculates an arbitration priority value, stored in APIC90 [Arbitration Priority (APR)], and the one with the lowest result accepts the interrupt.

The arbitration priority value is calculated by comparing APIC80[Priority] with the 8-bit encoded value of the highest bit set in APIC[270:200][RequestBits] (IRRVec) and the 8-bit encoded value of the highest bit set APIC[170:100][InServiceBits] (ISRVec). If APIC410[IerEn] is set the IRRVec and ISRVec are based off the highest enabled interrupt. The main priority bits[7:4] are compared as follows:

IF ((APIC80[Priority[7:4]] >= IRRVec[7:4]) && (APIC80[Priority[7:4]] > ISRVec[7:4])) THEN

```
    APIC90[Priority] = APIC80[Priority]
ELSEIF (IRRVec[7:4] > ISRVec[7:4]) THEN
    APIC90[Priority] = {IRRVec[7:4],0h}
ELSE
    APIC90[Priority] = {ISRVect[7:4],0h}
ENDIF.
```

2.4.9.1.12 Inter-Processor Interrupts

APIC300 [Interrupt Command Low (ICR Low)] and APIC310 [Interrupt Command High (ICR High)] provide a mechanism for generating interrupts in order to redirect an interrupt to another processor, originate an interrupt to another processor, or allow a processor to interrupt itself. A write to register APIC300 causes an interrupt to be generated with the properties specified by the APIC300 and APIC310 fields.

2.4.9.1.13 APIC Timer Operation

The local APIC contains a 32-bit timer, controlled by APIC320 [LVT Timer], APIC380 [Timer Initial Count], and APIC3E0 [Timer Divide Configuration]. The processor bus clock is divided by the value in APIC3E0[Div] to obtain a time base for the timer. When APIC380[Count] is written, the value is copied into APIC390 [Timer Current Count]. APIC390[Count] is decremented at the rate of the divided clock. When the count reaches 0, a timer interrupt is generated with the vector specified in APIC320[Vector]. If APIC320[Mode] specifies periodic operation, APIC390[Count] is reloaded with the APIC380[Count] value, and it continues to decrement at the rate of the divided clock. If APIC320[Mask] is set, timer interrupts are not generated.

2.4.9.1.14 Generalized Local Vector Table

All LVTs (APIC330 to APIC3[60:50], and APIC[530:500]) support a generalized message type as follows:

- 000b=Fixed
- 010b=SMI
- 100b=NMI
- 111b=ExtINT
- All other messages types are reserved.

2.4.9.1.15 State at Reset

At power-up or reset, the APIC is hardware disabled (MSR0000_001B[ApicEn]==0) so only SMI, NMI, INIT, and ExtInt interrupts may be accepted.

The APIC can be software disabled through APICF0[APICSWEn]. The software disable has no effect when the APIC is hardware disabled.

When a processor accepts an INIT interrupt, the APIC is reset as at power-up, with the exception that:

- APIC20[ApicId] is unaffected.
- Pending APIC register writes complete.

2.4.9.2 System Management Mode (SMM)

System management mode (SMM) is typically used for system control activities such as power management. These activities are typically transparent to the operating system.

2.4.9.2.1 SMM Overview

SMM is entered by a core on the next instruction boundary after a system management interrupt (SMI) is received and recognized. A core may be programmed to broadcast a special cycle to the system, indicating that it is entering SMM mode. The core then saves its state into the SMM memory state save area and jumps to the SMI service routine (or SMI handler). The pointer to the SMI handler is specified by MSRs. The code and data for the SMI handler are stored in the SMM memory area, which may be isolated from the main memory accesses.

The core returns from SMM by executing the RSM instruction from the SMI handler. The core restores its state from the SMM state save area and resumes execution of the instruction following the point where it entered SMM. The core may be programmed to broadcast a special bus cycle to the system, indicating that it is exiting SMM mode.

2.4.9.2.2 Operating Mode and Default Register Values

The software environment after entering SMM has the following characteristics:

- Addressing and operation is in Real mode.
 - A far jump, call or return in the SMI handler can only address the lower 1M of memory, unless the SMI handler first switches to protected mode.
 - If (MSRC001_0111[SmmBase]>=0010_0000h) then:
 - The value of the CS selector is undefined upon SMM entry.
 - The undefined CS selector value should not be used as the target of a far jump, call, or return.
- 4-Gbyte segment limits.
- Default 16-bit operand, address, and stack sizes (instruction prefixes can override these defaults).
- Control transfers that do not override the default operand size truncate the EIP to 16 bits.
- Far jumps or calls cannot transfer control to a segment with a base address requiring more than 20 bits, as in Real mode segment-base addressing, unless a change is made into protected mode.
- A20M# is disabled. A20M# assertion or deassertion have no affect during SMM.
- Interrupt vectors use the Real mode interrupt vector table.
- The IF flag in EFLAGS is cleared (INTR is not recognized).
- The TF flag in EFLAGS is cleared.
- The NMI and INIT interrupts are masked.
- Debug register DR7 is cleared (debug traps are disabled).

The SMM base address is specified by MSRC001_0111[SmmBase]. Important offsets to the base address pointer are:

- MSRC001_0111[SmmBase] + 8000h: SMI handler entry point.
- MSRC001_0111[SmmBase] + FE00h - FFFFh: SMM state save area.

2.4.9.2.3 SMI Sources And Delivery

The processor accepts SMIs as link-defined interrupt messages only. The core/node destination of these SMIs is a function of the destination field of these messages. However, the expectation is that all such SMI messages are specified to be delivered globally (to all cores of all nodes).

There are also several local events that can trigger SMIs. However, these local events do not generate SMIs directly. Each of them triggers a programmable IO cycle that is expected to target the SMI command port in the IO hub and trigger a global SMI interrupt message back to the coherent fabric.

Local sources of SMI events that generate the IO cycle specified in MSRC001_0056 [SMI Trigger IO Cycle]

are:

- In the core, as specified by:
 - MSRC001_0022 [Machine Check Exception Redirection].
 - MSRC001_00[53:50] [IO Trap (SMI_ON_IO_TRAP_[3:0])].
- All local APIC LVT registers programmed to generate SMIs.

The status for these is stored in SMMFEC4.

2.4.9.2.4 SMM Initial State

After storing the save state, execution starts at MSRC001_0111[SmmBase] + 08000h. The SMM initial state is specified in the following table.

Table 7: SMM Initial State

Register	SMM Initial State
CS	SmmBase[19:4]
DS	0000h
ES	0000h
FS	0000h
GS	0000h
SS	0000h
General-Purpose Registers	Unmodified
EFLAGS	0000_0002h
RIP	0000_0000_0000_8000h
CR0	Bits 0, 2, 3, and 31 cleared (PE, EM, TS, and PG); remainder is unmodified
CR4	0000_0000_0000_0000h
GDTR	Unmodified
LDTR	Unmodified
IDTR	Unmodified
TR	Unmodified
DR6	Unmodified
DR7	0000_0000_0000_0400h
EFER	All bits are cleared except bit 12 (SVME) which is unmodified.

2.4.9.2.5 SMM Save State

In the following table, the offset field provides the offset from the SMM base address specified by MSRC001_0111 [SMM Base Address (SMM_BASE)].

Table 8: SMM Save State

Offset	Size	Contents		Access
FE00h	Word	ES	Selector	Read-only
FE02h	6 Bytes		Reserved	
FE08h	Quadword		Descriptor in memory format	

Table 8: SMM Save State (Continued)

Offset	Size	Contents		Access
FE10h	Word	CS	Selector	Read-only
FE12h	6 Bytes		Reserved	
FE18h	Quadword		Descriptor in memory format	
FE20h	Word	SS	Selector	Read-only
FE22h	6 Bytes		Reserved	
FE28h	Quadword		Descriptor in memory format	
FE30h	Word	DS	Selector	Read-only
FE32h	6 Bytes		Reserved	
FE38h	Quadword		Descriptor in memory format	
FE40h	Word	FS	Selector	Read-only
FE42h	2 Bytes		Reserved	
FE44h	Doubleword		FS Base {16'b[47], 47:32}[1]	
FE48h	Quadword		Descriptor in memory format	
FE50h	Word	GS	Selector	Read-only
FE52h	2 Bytes		Reserved	
FE54h	Doubleword		GS Base {16'b[47], 47:32}[1]	
FE58h	Quadword		Descriptor in memory format	
FE60h	4 Bytes	GDTR	Reserved	Read-only
FE64h	Word		Limit	
FE66h	2 Bytes		Reserved	
FE68h	Quadword		Descriptor in memory format	
FE70h	Word	LDTR	Selector	Read-only
FE72h	Word		Attributes	
FE74h	Doubleword		Limit	
FE78h	Quadword		Base	
FE80h	4 Bytes	IDTR	Reserved	Read-only
FE84h	Word		Limit	
FE86h	2 Bytes		Reserved	
FE88h	Quadword		Base	
FE90h	Word	TR	Selector	Read-only
FE92h	Word		Attributes	
FE94h	Doubleword		Limit	
FE98h	Quadword		Base	
FEA0h	Quadword	IO_RESTART_RIP		Read-only
FEA8h	Quadword	IO_RESTART_RCX		
FEB0h	Quadword	IO_RESTART_RSI		
FEB8h	Quadword	IO_RESTART_RDI		
FEC0h	Doubleword	SMMFEC0 [SMM IO Trap Offset]		Read-only
FEC4	Doubleword	SMMFEC4 [Local SMI Status]		Read-only

Table 8: SMM Save State (Continued)

Offset	Size	Contents	Access
FEC8h	Byte	SMMFEC8 [SMM IO Restart Byte]	Read-write
FEC9h	Byte	SMMFEC9 [Auto Halt Restart Offset]	Read-write
FECAh	Byte	SMMFECA [NMI Mask]	Read-write
FECBh	5 Bytes	Reserved	
FED0h	Quadword	EFER	Read-only
FED8h	Quadword	SMMFED8 [SMM SVM State]	Read-only
FEE0h	Quadword	Guest VMCB physical address	Read-only
FEE8h	Quadword	SVM Virtual Interrupt Control	Read-only
FEF0h	16 Bytes	Reserved	
FEFCh	Doubleword	SMMFEFC [SMM-Revision Identifier]	Read-only
FF00h	Doubleword	SMMFF00 [SMM Base Address (SMM_BASE)]	Read-write
FF04h	28 Bytes	Reserved	
FF20h	Quadword	Guest PAT	Read-only
FF28h	Quadword	Host EFER[2]	
FF30h	Quadword	Host CR4[2]	
FF38h	Quadword	Nested CR3[2]	
FF40h	Quadword	Host CR0[2]	
FF48h	Quadword	CR4	
FF50h	Quadword	CR3	
FF58h	Quadword	CR0	
FF60h	Quadword	DR7	Read-only
FF68h	Quadword	DR6	
FF70h	Quadword	RFLAGS	Read-write
FF78h	Quadword	RIP	Read-write
FF80h	Quadword	R15	
FF88h	Quadword	R14	
FF90h	Quadword	R13	
FF98h	Quadword	R12	
FFA0h	Quadword	R11	
FFA8h	Quadword	R10	
FFB0h	Quadword	R9	
FFB8h	Quadword	R8	

Table 8: SMM Save State (Continued)

Offset	Size	Contents	Access
FFC0h	Quadword	RDI	Read-write
FFC8h	Quadword	RSI	
FFD0h	Quadword	RBP	
FFD8h	Quadword	RSP	
FFE0h	Quadword	RBX	
FFE8h	Quadword	RDX	
FFF0h	Quadword	RCX	
FFF8h	Quadword	RAX	

Notes:
1. This notation specifies that bit[47] is replicated in each of the 16 MSBs of the DW (sometimes called *sign extended*). The 16 LSBs contain bits[47:32].
2. Only used for an SMI in guest mode with nested paging enabled.

The SMI save state includes most of the integer execution unit. Not included in the save state are: the floating point state, MSRs, and CR2. In order to be used by the SMI handler, these must be saved and restored. The save state is the same, regardless of the operating mode (32-bit or 64-bit).

The following are some offsets in the SMM save state area. The mnemonic for each offset is in the form SMMxxxx, where xxxx is the offset in the save state.

SMMFEC0 SMM IO Trap Offset

If the assertion of SMI is recognized on the boundary of an IO instruction, SMMFEC0 [SMM IO Trap Offset] contains information about that IO instruction. For example, if an IO access targets an unavailable device, the system can assert SMI and trap the IO instruction. SMMFEC0 then provides the SMI handler with information about the IO instruction that caused the trap. After the SMI handler takes the appropriate action, it can reconstruct and then re-execute the IO instruction from SMM. Or, more likely, it can use SMMFEC8 [SMM IO Restart Byte], to cause the core to re-execute the IO instruction immediately after resuming from SMM.

Bits	Description
31:16	**Port: trapped IO port address**. Read-only. This provides the address of the IO instruction.
15:12	**BPR: IO breakpoint match**. Read-only.
11	**TF:** EFLAGS TF value. Read-only.
10:7	Reserved.
6	**SZ32: size 32 bits**. Read-only. 1=Port access was 32 bits.
5	**SZ16: size 16 bits**. Read-only. 1=Port access was 16 bits.
4	**SZ8: size 8 bits**. Read-only. 1=Port access was 8 bits.
3	**REP: repeated port access**. Read-only.
2	**STR: string-based port access**. Read-only.
1	**V: IO trap word valid**. Read-only. 1=The core entered SMM on an IO instruction boundary; all information in this offset is valid. 0=The other fields of this offset are not valid.
0	**RW: port access type**. Read-only. 0=IO write (OUT instruction). 1=IO read (IN instruction).

SMMFEC4 Local SMI Status

This offset stores status bits associated with SMI sources local to the core. For each of these bits, 1=The associated mechanism generated an SMI.

Bits	Description
31:20	Reserved.
19	**SmiSrcThrCntHt: SMI source link thresholding**. Read-only. This bit is associated with the SMI source specified in the link thresholding register (see MSR0000_0403 [MC0 Machine Check Miscellaneous (MC0_MISC)]).
18	**SmiSrcThrCntDram: SMI source DRAM thresholding**. Read-only. This bit is associated with the SMI source specified in the DRAM thresholding register (see D18F3x160 [NB Machine Check Misc (DRAM Thresholding) 0 (MC4_MISC0)]).
17	**SmiSrcLvtExt: SMI source LVT extended entry**. Read-only. This bit is associated with the SMI sources specified in APIC[530:500] [Extended Interrupt [3:0] Local Vector Table].
16	**SmiSrcLvtLcy: SMI source LVT legacy entry**. Read-only. This bit is associated with the SMI sources specified by the non-extended LVT entries of the APIC.
15:11	Reserved.
10	**IntPendSmiSts: interrupt pending SMI status**. Read-only. This bit is associated with the SMI source specified when (MSRC001_0055[IntPndMsg] == 1).
9	Reserved.
8	**MceRedirSts: machine check exception redirection status**. Read-only. This bit is associated with the SMI source specified in MSRC001_0022[RedirSmiEn].
7:4	Reserved.
3:0	**IoTrapSts: IO trap status**. Read-only. Each of these bits is associated with each of the respective SMI sources specified in MSRC001_00[53:50] [IO Trap (SMI_ON_IO_TRAP_[3:0])].

SMMFEC8 SMM IO Restart Byte

00h on entry into SMM.

If the core entered SMM on an IO instruction boundary, the SMI handler may write this to FFh. This causes the core to re-execute the trapped IO instruction immediately after resuming from SMM. The SMI handler should only write to this byte if SMMFEC0 field V == 1; otherwise, the behavior is undefined.

If a second SMI is asserted while a valid IO instruction is trapped by the first SMI handler, the core services the second SMI prior to re-executing the trapped IO instruction. SMMFEC0 field V == 0 during the second entry into SMM, and the second SMI handler must not rewrite this byte.

If there is a simultaneous SMI IO instruction trap and debug breakpoint trap, the processor first responds to the SMI and postpones recognizing the debug exception until after resuming from SMM. If debug registers other than DR6 and DR7 are used while in SMM, they must be saved and restored by the SMI handler. If SMMFEC8 [SMM IO Restart Byte], is set to FFh when the RSM instruction is executed, the debug trap does not occur until after the IO instruction is re-executed.

Bits	Description
7:0	**RST: SMM IO Restart Byte**. Read-write.

SMMFEC9 Auto Halt Restart Offset

Bits	Description
7:1	Reserved.
0	**HLT: halt restart**. Read-write. Upon SMM entry, this bit indicates whether SMM was entered from the Halt state. 0=Entered SMM on a normal x86 instruction boundary. 1=Entered SMM from the Halt state. Before returning from SMM, this bit can be written by the SMI handler to specify whether the return from SMM should take the processor back to the Halt state or to the instruction-execution state specified by the SMM state save area (normally, the instruction after the halt). 0=Return to the instruction specified in the SMM save state. 1=Return to the Halt state. If the return from SMM takes the processor back to the Halt state, the HLT instruction is not refetched and re-executed. However, the Halt special bus cycle is broadcast and the processor enters the Halt state.

SMMFECA NMI Mask

Bits	Description
7:1	Reserved.
0	**NmiMask**. Read-write. Specifies whether NMI was masked upon entry to SMM. 0=NMI not masked. 1=NMI masked.

SMMFED8 SMM SVM State

Read-only. This offset stores the SVM state of the processor upon entry into SMM.

Bits	Description
63:4	Reserved.
3	**HostEflagsIf: host** EFLAGS IF.
2:0	**SvmState**. Bits — Definition 000b — SMM entered from a non-guest state. 001b — Reserved. 010b — SMM entered from a guest state. 101b-011b — Reserved. 110b — SMM entered from a guest state with nested paging enabled. 111b — Reserved.

SMMFEFC SMM-Revision Identifier

SMM entry state: 0003_0064h.

Bits	Description
31:18	Reserved.
17	**BRL**. Read-only. Base relocation supported.
16	**IOTrap**. Read-only. IO trap supported.
15:0	**Revision**. Read-only.

SMMFF00 SMM Base Address (SMM_BASE)

Bits	Description
31:0	See: MSRC001_0111[SmmBase].

2.4.9.2.6 Exceptions and Interrupts in SMM

When SMM is entered, the core masks INTR, NMI, SMI, INIT, and A20M interrupts. The core clears the IF flag to disable INTR interrupts. To enable INTR interrupts within SMM, the SMM handler must set the IF flag to 1. A20M is disabled so that address bit[20] is never masked when in SMM.

Generating an INTR interrupt can be used for unmasking NMI interrupts in SMM. The core recognizes the assertion of NMI within SMM immediately after the completion of an IRET instruction. Once NMI is recognized within SMM, NMI recognition remains enabled until SMM is exited, at which point NMI masking is restored to the state it was in before entering SMM.

While in SMM, the core responds to the DBREQ and STPCLK interrupts, as well as to all exceptions that may be caused by the SMI handler.

2.4.9.2.7 The Protected ASeg and TSeg Areas

These ranges are controlled by MSRC001_0112 and MSRC001_0113; see those registers for details.

2.4.9.2.8 SMM Special Cycles

Special cycles can be initiated on entry and exit from SMM to acknowledge to the system that these transitions are occurring. These are controlled by MSRC001_0015[RsmSpCycDis, SmiSpCycDis].

2.4.9.2.9 Locking SMM

The SMM registers (MSRC001_0112 and MSRC001_0113) can be locked from being altered by setting MSRC001_0015[SmmLock]. SBIOS must lock the SMM registers after initialization to prevent unexpected changes to these registers.

2.4.9.2.10 Synchronizing SMM Entry (Spring-Boarding)

The BIOS must take special care to ensure that all cores have entered SMM prior to accessing shared IO resources and all core SMI interrupt status bits are synchronized. This generally requires that BIOS waits for

all cores to enter SMM.

The following conditions can cause one or more cores to enter SMM without all cores entering SMM:

- More than one IO device in the system is enabled to signal an SMI without hardware synchronization (e.g., using an end of SMI gate).
- A single device may signal multiple SMI messages without hardware synchronization (e.g., using an end of SMI gate).
- An SMI is received while one or more AP cores are in the INIT state. This may occur either during BIOS or secure boot.
- A hardware error prevents a core from entering SMM.

The act of synchronizing cores into SMM is called spring-boarding. Because not all of the above conditions can be avoided, it is recommended that all systems support spring-boarding.

An ACPI-compliant IO hub is required for spring-boarding. Depending on the IO hub design, BIOS may have to set additional end-of-SMI bits to trigger an SMI from within SMM.

The software requirements for the suggested spring-boarding implementation are listed as follows.

- A binary semaphore located in SMRAM, accessible by all cores. For the purpose of this discussion, the semaphore is called CheckSpringBoard. CheckSpringBoard is initialized to zero.
- Two semaphores located in SMRAM, accessible by all cores. For the purpose of this discussion, the semaphores are called NotInSMM and WaitInSMM. NotInSMM and WaitInSMM are initialized to a value equal to the number of cores in the system (NumCPUs).

The following BIOS algorithm describes spring-boarding and is optimized to reduce unnecessary SMI activity. This algorithm must be made part of the SMM instruction sequence for each core in the system.

1. Attempt to obtain ownership of the CheckSpringBoard semaphore with a read-modify-write instruction. If ownership was obtained then do the following, else proceed to step 2:
 - Check all enabled SMI status bits in the IO hub.
 Let Status=enable1&status1 | enable2&status2 | enable3&status3 … enable n & status n.
 - If (Status==0) then perform the following sub-actions.
 - Trigger an SMI broadcast assertion from the IO hub by writing to the software SMI command port.
 - Resume from SMM with the RSM instruction.

 //Example:

```
InLineASM{
   BTS CheckSpringBoard,0; Try to obtain ownership of semaphore
   JC Step_2:
   CALL CheckIOHUB_SMIEVT; proc returns ZF=1 for no events
   JNZ Step_2:
   CALL Do_SpringBoard;Trigger SMI and then RSM
Step_2:
}
```

2. Decrement the NotInSMM variable. Wait for (NotInSMM == 0). See Note 1.
3. Execute the core-local event SMI handler. Using a third semaphore (not described here), synchronize core execution at the end of the task. After all cores have executed, proceed to step 4. The following is a brief description of the task for each core:
 - Check all enabled core-local SMI status bits in the core's private or MSR address space. Handle the event if possible, or pass information necessary to handle the event to a mailbox for the BSC to handle.
 - An exclusive mailbox must exist for each core for each core local event.
 - Wait for all cores to complete this task at least once.

4. If the current core executing instructions is not the BSC then jump to step 5. If the core executing instructions is the BSC then jump to the modified main SMI handler task, described below.
 - Check all enabled SMI status bits in the IO hub. Check mailboxes for event status.
 - For each event, handle the event and clear the corresponding status bit.
 - Repeat until all enabled SMI status bits are clear and no mailbox events remain.
 - Set NotInSMM=NumCPUs. (Jump to step 5.)
5. Decrement the WaitInSMM variable. Wait for WaitInSMM=0. See Note 2.
6. Increment the WaitInSMM variable. Wait for WaitInSMM=NumCPUs.
7. If the current processor core executing instructions is the BSC then reset CheckSpringBoard to zero.
8. Resume from SMM with the RSM instruction.

Notes:
1. To support a secure startup by the secure loader the BIOS must provide a timeout escape from the otherwise endless loop. The timeout value should be large enough to account for the latency of all cores entering SMM. The maximum SMM entrance latency is defined by the platform's IO sub-system, not the processor. A value of twice the watchdog timer count is recommended. See D18F3x44 [MCA NB Configuration] for more information on the watchdog time-out value. If a time-out occurs in the wait loop, the BIOS (the last core to decrement NotInSMM) should record the number of cores that have not entered SMM and all cores must fall out of the loop.
2. If a time-out occurs in the wait loop in step 2, the BIOS must not wait for WaitInSMM = 0. Instead it must wait for WaitInSMM = (the number of cores recorded in step 2).
3. If BIOS places APs in the INIT state during any part of the boot process when SMIs may be generated, or may generate SMIs before taking all APs out of their initial microcode reset loop (i.e., before D18F0x1DC[CpuEn] is set), then it is recommended that BIOS keep a record of how many APs are in these two states and exclude these cores from the wait loops. SMIs are not recognized by a processor in these states. AMD does not recommend enabling SMI sources prior to bringing all APs out of these states.

2.4.10 Secure Virtual Machine Mode (SVM)

Support for SVM mode is indicated by CPUID Fn8000_0001_ECX[SVM].

2.4.10.1 BIOS support for SVM Disable

The BIOS should include the following user setup options to enable and disable AMD Virtualization™ technology.

- Enable AMD Virtualization™:
 - MSRC001_0114[SvmeDisable] = 0.
 - MSRC001_0114[Lock] = 1.
 - MSRC001_0118[SvmLockKey] = 0000_0000_0000_0000h.
- Disable AMD Virtualization™:
 - MSRC001_0114[SvmeDisable] = 1.
 - MSRC001_0114[Lock] = 1.
 - MSRC001_0118[SvmLockKey] = 0000_0000_0000_0000h.

The BIOS may also include the following user setup options to disable AMD Virtualization™ technology.

- Disable AMD Virtualization™, with a user supplied key:
 - MSRC001_0114[SvmeDisable] = 1.
 - MSRC001_0114[Lock] = 1.
 - MSRC001_0118[SvmLockKey] programmed with value supplied by user. This value should be stored

in NVRAM.

2.4.11 CPUID Instruction

The CPUID instruction provides data about the features supported by the processor. See 3.19 [CPUID Instruction Registers].

2.4.11.1 Multi-Core Support

There are two methods for determining multi-core support. A recommended mechanism is provided and a legacy method is also available for existing operating systems. System software should use the correct architectural mechanism to detect the number of physical cores by observing CPUID Fn8000_0008_ECX[NC]. The legacy method utilizes the CPUID Fn0000_0001_EBX[LogicalProcessorCount].

2.5 Power Management

The processor supports a wide variety of power management features, including:
- OS-directed power management such as ACPI.
- Clock frequency and voltage states (refered to as P-states and DPM states), including:
 - CPU core P-states.
 - Northbridge P-states.
 - Memory P-states.
 - Graphics DPM states.
 - Multi-media block DPM states.
- Power and thermal management for performance.
 - Power optimization between blocks for optimal performance.
 - Voltage transient tolerance.
- Power efficiency for battery life, including:
 - Power gating.
 - Voltage optimization.
 - Deep sleep modes (e.g., ACPI S3, or connected standby S0i3).
 - Limiting frequency when it provides little value.
- BIOS-configurable specifications.

2.5.1 Processor Power Planes And Voltage Control

Refer to the *Electrical Data Sheet* for power plane definitions. See 1.2 [Reference Documents].

2.5.1.1 Serial VID Interface

The processor includes an interface to control external voltage regulators, called the serial VID interface (SVI). The frequency of SVC for SVI2 is defined in the SVI2 specification. See the *AMD Serial VID Interface 2.0 (SVI2) Specification* for additional details.

The processor supports two SVI interfaces that are allocated to power planes as follows:

Table 9: Serial VID Interface (SVI) Port Mapping to SOC Power Planes

Port	SVI 2.0 specification “VDD” plane connected to:	SVI 2.0 specification “VDDNB” plane connected to:
SVI0	VDD	VDDNB
SVI1	VDDGFX	Not connected

2.5.1.1.1 SVI2 Features

The processor supports the following SVI2 features:

- Voltage offsets:
 - VDD: Core load line offset.
 - VDDNB: NB load line offset.

- Load line trim:
 - VDD: Core load line trim.
 - VDDNB: NB load line trim.

2.5.1.2 Internal VID Registers and Encodings

All VID register fields within the processor are 8-bits wide. See the *AMD Serial VID Interface 2.0 (SVI2) Specification* for additional details.

The boot VID is 0.9 volts.

2.5.1.2.1 MinVid and MaxVid Check

Hardware limits the minimum and maximum VID code that is sent to the voltage regulator. Prior to generating VID-change commands to SVI, the processor filters the InputVid value to the OutputVid as follows (higher VID codes correspond to lower voltages and lower VID codes correspond to higher voltages):

- If InputVid < MAX_VID, OutputVid=MAX_VID.
 - Else if (InputVid > MIN_VID) & (MIN_VID!=00h), OutputVid=MIN_VID.
 - Else OutputVid=InputVid.

This filtering is applied regardless of the source of the VID-change command.

2.5.1.3 Low Power Features

2.5.1.3.1 PSIx_L Bit

The processor can indicate whether or not it's in a low-voltage state via the PSIx_L bit. This indicator may be used by the voltage regulator to place itself into a more power efficient mode. PSIx_L is controlled independently for VDD and VDDNB.

- The processor supports the PSI0_L and the PSI1_L bits in the data fields of the SVI2 command.
 - Changes to the state of PSI0_L can only occur on VID changes.
 - Changes to the state of PSI1_L can occur at any time.

2.5.1.3.1.1 BIOS Requirements for PSI0_L

Enabling PSI0_L for the VDD and VDDNB planes depends on support from the voltage regulator and is therefore system specific. The voltage regulator must be able to supply the current required for the processor to operate at the VID code specified. Depending on the regulator used, AMD recommends one of the following methods:

- PSI0_L disabled:
 - VDD: Set PSI0_L high for the VDD plane.
 - VDDNB: Set PSI0_L high for the VDDNB plane.

- PSI0_L set/clear based on current requirements:
 - VDD: The following algorithm describes how to program PSI0_L on VDD:

```
PSI_vrm_current = current at which the regulator allows PSI0_L.
previous_voltage = FFh

for (each P-state from P0 to D18F3xDC[HwPstateMaxVal]) {
  pstate_current = ProcIddMax for the current P-state,
                   see 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check];
  pstate_voltage = MSRC001_00[6B:64][CpuVid] of the current P-state;

  if (current P-state == D18F3xDC[HwPstateMaxVal]) {
```

```
    next_pstate_current = 0;
  } else {
    next_pstate_current = ProcIddMax for the next P-state,
                  see 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check];
  }

  if ((pstate_current <= PSI_vrm_current) &&
      (next_pstate_current <= PSI_vrm_current) &&
      (pstate_voltage != previous_voltage)) {
    Program PSI_L VID threshold = pstate_voltage;
    Enable control over PSI_L;
    break;
  }
  previous_voltage = pstate_voltage;
}
```

- VDDNB: The following algorithm describes how to program PSI0_L on VDDNB:

 NbIddMax = D18F5x16[C:0][NbIddDiv] current.

```
PSI_vrm_current = current at which the VDDNB regulator allows PSI0_L.
previous_voltage = FFh

for (each valid NB P-state starting with NBP0) {
  pstate_current = NbIddMax of the current NB P-state;
  pstate_voltage = D18F5x16[C:0][NbVid] of the current NB P-state;

  if (current P-state is the last valid P-state) {
    next_pstate_current = 0;
  } else {
    next_pstate_current = NbIddMax for the next P-state;
  }

  if ((pstate_current <= PSI_vrm_current) &&
      (next_pstate_current <= PSI_vrm_current) &&
      (pstate_voltage != previous_voltage)) {
    Program NB PSI0_L VID threshold = pstate_voltage;
    Enable control over NB PSI0_L;
    break;
  }
  previous_voltage = pstate_voltage;
}
```

2.5.1.4 Voltage Transitions

The processor supports dynamic voltage transitions on SVI-controlled power planes. These transitions are managed during state changes such as reset, P-state changes, and C-state changes. See the *AMD Serial VID Interface 2.0 (SVI2) Specification* for additional details about how voltage transitions occur.

Notes about voltage-frequency interaction:

- If a voltage increase is requested, the processor waits the amount of time specified before sending any additional voltage change requests to the voltage regulator or before beginning a frequency transition.
- If a voltage decrease is requested, the processor waits the amount of time specified by D18F5x128[FastSlamTimeDown] before sending any additional voltage change requests to the voltage regulator. For voltage decreases, the processor immediately changes frequency, before the voltage change starts.

The code execution is allowed during voltage transitions.

2.5.2 CPU Core Power Management

2.5.2.1 Core P-states

Core P-states are operational performance states characterized by a unique combination of core frequency and voltage. The processor supports up to 8 core P-states (P0 through P7), specified in MSRC001_00[6B:64]. Out of cold reset, the voltage and frequency of the compute units is specified by MSRC001_0071[StartupPstate].

Support for dynamic core P-state changes is indicated by more than one enabled selection in MSRC001_00[6B:64][PstateEn]. Software requests core P-state changes for each core independently. Support for hardware P-state control is indicated by CPUID Fn8000_0007_EDX[HwPstate]=1b. Software may not request any P-state transitions until the P-state initialization requirements defined in 2.5.2.1.5 [BIOS Requirements for Core P-state Initialization and Transitions] are complete.

The processor supports independently-controllable frequency planes for each compute unit and independently-controllable voltage plane for all compute units.

The following terms apply to each of these planes:

- FID: frequency ID. Specifies the PLL frequency multiplier, relative to the reference clock, for a given domain.
- DID: divisor ID. Specifies the post-PLL power-of-two divisor that can be used to reduce the operating frequency.
- COF: current operating frequency. Specifies the operating frequency as a function of the FID and DID. Refer to CoreCOF for the CPU COF formula and NBCOF for the NB COF formula.
- VID: voltage ID. Specifies the voltage level for a given power plane.

2.5.2.1.1 Core P-state Naming and Numbering

The number of boosted P-states may vary from product to product. Thus, the mapping between MSRC001_00[6B:64] and the indices used to request P-state changes or status also varies. In order to clarify this, two different numbering schemes are used.

2.5.2.1.1.1 Software P-state Numbering

When referring to software P-state numbering, the following naming convention is used:

- Non-boosted P-states are referred to as P0, P1, etc.
 - Software P0 is the highest power, highest performance, non-boosted P-state.
 - Software P0 is also referred to as the base P-state.
 - Each ascending P-state number represents a lower-power, lower performance non-boosted P-state than the prior P-state number.
- Boosted P-states are referred to as Pb0, Pb1, etc.
 - Pb0 is the highest-performance, highest-power boosted P-state.
 - Each higher numbered boosted P-state represents a lower-power, lower-performance boosted P-state.

For example, if D18F4x15C[NumBoostStates] contains the values shown below, then the P-states would be named as follows:

Table 10: Software P-state Naming

D18F4x15C[NumBoostStates]=1		D18F4x15C[NumBoostStates]=3	
P-state Name	**Corresponding Register Address**	**P-state Name**	**Corresponding Register Address**
Pb0	MSRC001_0064	Pb0	MSRC001_0064
P0 (base)	MSRC001_0065	Pb1	MSRC001_0065
P1	MSRC001_0066	Pb2	MSRC001_0066
P2	MSRC001_0067	P0 (base)	MSRC001_0067
P3	MSRC001_0068	P1	MSRC001_0068
P4	MSRC001_0069	P2	MSRC001_0069
P5	MSRC001_006A	P3	MSRC001_006A
P6	MSRC001_006B	P4	MSRC001_006B

All sections and register definitions use software P-state numbering unless otherwise specified.

2.5.2.1.1.2 Hardware P-state Numbering

When referring to hardware P-state numbering, the following naming convention is used:

- All P-states are referred to as P0, P1, etc.
 - P0 is the highest power, highest-performance P-state, regardless of whether it is a boosted P-state or a non-boosted P-state.
 - Each ascending P-state number represents a lower-power, lower-performance P-state, regardless of whether it is a boosted P-state or not.

2.5.2.1.2 Core P-state Control

Core P-states are dynamically controlled by software and are exposed through ACPI objects (refer to 2.5.2.1.7.3 [ACPI Processor P-state Objects]). Software requests a core P-state change by writing a 3 bit index corresponding to the desired P-state number to MSRC001_0062[PstateCmd] of the appropriate core. For example, to request P3 for core 0 software would write 011b to core 0's MSRC001_0062[PstateCmd].

2.5.2.1.3 Core P-state Visibility

MSRC001_0063[CurPstate] reflects the current non-boosted P-state number for each compute unit. For example, if MSRC001_0063[CurPstate]=010b on compute unit 1, then compute unit 1 is in the P2 state. If a compute unit is in a boosted P-state, MSRC001_0063[CurPstate] reads back as 0.

2.5.2.1.4 Core P-state Limits

Core P-states may be limited to lower-performance values under certain conditions, including:

- HTC. See D0F0xBC_xD820_0C64[HTC_PSTATE_LIMIT].
- Software. See D18F3x68[SwPstateLimit].
- PROCHOT_L assertion. See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].
- Other power management functions.

P-state limits are applied to all cores on the processor. The current P-state limit is provided in MSRC001_0061[CurPstateLimit]. Changes to the MSRC001_0061[CurPstateLimit] can be programmed to

trigger interrupts through D0F0xBC_xD820_0C64[HTC_APIC_LO_EN, HTC_APIC_HI_EN]. In addition, the maximum P-state value, regardless of the source, is limited as specified in MSRC001_0061[PstateMaxVal].

2.5.2.1.5 BIOS Requirements for Core P-state Initialization and Transitions

1. Check that CPUID Fn8000_0007_EDX[HwPstate]=1. If not, P-states are not supported, no P-state related ACPI objects should be generated, and BIOS must skip the rest of these steps.
2. Ensure the following fields are configured to their BIOS recommendations:
 - D0F0xBC_xD822_20B8[Pll_LOCK_TIMER].
 - D0F0xBC_xD822_2114[GatersOnTime, GatersOffTime].
3. Transition all cores to the minimum performance P-state using the algorithm detailed in 2.5.2.1.7.2 [Core Minimum P-state Transition Sequence After Warm Reset].
4. Complete the 2.5.3.1.2.1 [NB P-state COF and VID Synchronization After Warm Reset]. All cores on a processor must be in the minimum performance P-state prior to executing this sequence.
5. Complete the 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check].
6. Perform the following steps in any order:
 A. Enable 2.5.8 [Application Power Management (APM)] as follows:
 - Ensure the following fields are configured to their BIOS recommendations:
 - D18F4x110[CSampleTimer].
 - D18F4x15C[ApmMasterEn].
 - D18F5xE0[RunAvgRange].
 - See your AMD representative for details on how to enable the GPU aspects of 2.5.8 [Application Power Management (APM)].
 - If D18F4x15C[NumBoostStates]!=0, program D18F4x15C[BoostSrc]=1.
 B. Transition all cores to the maximum performance P-state by writing 0 to MSRC001_0062[PstateCmd].
 C. Create ACPI objects if neccessary:
 - Determine the valid set of P-states as indicated by MSRC001_00[6B:64][PstateEn].
 - If P-states are not supported, as indicated by only one enabled selection in MSRC001_00[6B:64][PstateEn], then BIOS must not generate ACPI-defined P-state objects described in 2.5.2.1.7.3 [ACPI Processor P-state Objects]. Otherwise, the ACPI objects should be generated to enable P-state support.
 D. Configure the COF and VID for each processor appropriately based on the sequence described in 2.5.3.1.2 [BIOS NB P-state Configuration].
7. Configure PSIx_L. Refer to 2.5.1.3.1 [PSIx_L Bit] for additional details.

2.5.2.1.6 Processor-Systemboard Current Delivery Compatibility Check

P-states that require higher electrical design current (EDC) delivery than the systemboard is specified to provide are disabled automatically by the processor. To do this, the processor calculates the maximum current required by each P-state as follows:

ProcIddMax = MSRC001_00[6B:64][IddValue] current * 1/10^MSRC001_00[6B:64][IddDiv] * (D18F5x84[CmpCap]+1)

These values are compared to the platform-specified EDC and any that exceed that value are not utilized by the processor.

2.5.2.1.7 BIOS COF and VID Requirements After Warm Reset

Warm reset is asynchronous and can interrupt P-state transitions leaving the processor in a VID state that does

not correspond to MSRC001_0063[CurPstate] on any core. The processor frequency after warm reset corresponds to MSRC001_0063[CurPstate]. BIOS is required to transition the processor to valid COF and VID settings corresponding to an enabled P-state following warm reset. The cores may be transitioned to either the maximum or minimum P-state COF and VID settings using the sequences defined in 2.5.2.1.7.1 [Core Maximum P-state Transition Sequence After Warm Reset] and 2.5.2.1.7.2 [Core Minimum P-state Transition Sequence After Warm Reset]. Transitioning to the minimum P-state after warm reset is recommended to prevent undesired system behavior if a warm reset occurs before the 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check] is complete. BIOS is not required to manipulate NB COF and VID settings following warm reset if the warm reset was issued by BIOS to update D18F5x16[C:0][NbFid].

2.5.2.1.7.1 Core Maximum P-state Transition Sequence After Warm Reset

1. If MSRC001_0071[CurPstate] = D18F3xDC[HwPstateMaxVal], then skip step 3 for that core.
2. Write MSRC001_0061[PstateMaxVal] to MSRC001_0062[PstateCmd] on all cores in the processor.
3. Wait for MSRC001_0071[CurCpuFid, CurCpuDid] = [CpuFid[5:0], CpuDid] from MSRC001_00[6B:64] indexed by D18F3xDC[HwPstateMaxVal].
4. All previous steps must be completed on all cores prior to continuing the sequence since a compute unit transitions to the highest performance P-state requested on either core.
5. Write 0 to MSRC001_0062[PstateCmd] on all cores in the processor.
6. Wait for MSRC001_0071[CurCpuFid, CurCpuDid] = [CpuFid[5:0], CpuDid] from MSRC001_00[6B:64] indexed by MSRC001_0071[CurPstateLimit].
7. If MSRC001_0071[CurPstateLimit] != D18F3xDC[HwPstateMaxVal], wait for MSRC001_0071[CurCpuVid] = [CpuVid] from MSRC001_00[6B:64] indexed by MSRC001_0071[CurPstateLimit].
8. Wait for MSRC001_0063[CurPstate] = MSRC001_0061[CurPstateLimit].

2.5.2.1.7.2 Core Minimum P-state Transition Sequence After Warm Reset

1. If MSRC001_0071[CurPstate] = MSRC001_0071[CurPstateLimit], then skip step 3 for that core.
2. Write 0 to MSRC001_0062[PstateCmd] on all cores in the processor.
3. Wait for MSRC001_0071[CurCpuFid, CurCpuDid] = [CpuFid[5:0], CpuDid] from MSRC001_00[6B:64] indexed by MSRC001_0071[CurPstateLimit].
4. Write MSRC001_0061[PstateMaxVal] to MSRC001_0062[PstateCmd] on all cores in the processor.
5. Wait for MSRC001_0071[CurCpuFid, CurCpuDid] = [CpuFid[5:0], CpuDid] from MSRC001_00[6B:64] indexed by D18F3xDC[HwPstateMaxVal].
6. If MSRC001_0071[CurPstateLimit] != MSRC001_0071[CurPstate], wait for MSRC001_0071[CurCpuVid] = [CpuVid] from MSRC001_00[6B:64] indexed by D18F3xDC[HwPstateMaxVal].
7. Wait for MSRC001_0063[CurPstate] = MSRC001_0062[PstateCmd].

2.5.2.1.7.3 ACPI Processor P-state Objects

Processor performance control is implemented through the _PCT, _PSS and _PSD objects in ACPI 2.0 and later revisions. The presence of these objects indicates to the OS that the platform and processor are capable of supporting multiple performance states. Processor performance states are not supported with ACPI 1.0b. BIOS must provide the _PCT, _PSS, and _PSD objects, and define other ACPI parameters to support operating systems that provide native support for processor P-state transitions.

The following rules apply to BIOS generated ACPI objects in multi-core systems. Refer to the appropriate ACPI specification for additional details:

- All cores must expose the same number of performance states to the OS.
- The respective performance states displayed to the OS for each core must have identical performance and power-consumption parameters (e.g., P0 on core 0 must have the same performance and power-consumptions parameters as P0 on core 1, P1 on core 0 must have the same parameters as P1 on core 1, however P0 can be different than P1).
- Performance state objects must be present under each processor object in the system.

2.5.2.1.7.3.1 _PCT (Performance Control)

BIOS must declare the performance control object parameters as functional fixed hardware. This definition indicates the processor driver understands the architectural definition of the P-state interface associated with CPUID Fn8000_0007_EDX[HwPstate]=1.

- Perf_Ctrl_Register = Functional Fixed Hardware
- Perf_Status_Register = Functional Fixed Hardware

2.5.2.1.7.3.2 _PSS (Performance Supported States)

A unique _PSS entry is created for each non-boosted P-state. The value contained in the _PSS Control field is written to MSRC001_0062 [P-state Control] to request a P-state change to the CoreFreq of the associated _PSS object. The value contained in MSRC001_0063 [P-state Status] can be used to identify the _PSS object of the current P-state request by equating MSRC001_0063[CurPstate] to the value of the Status field. See 2.5.2.1 [Core P-states].

BIOS loops through each of MSRC001_00[6B:64] applying the following formulas to create the fields for the _PSS object for for each valid P-state (see MSRC001_00[6B:64][PstateEn]). BIOS skips over any P-state MSRs that specify boost P-states (see D18F4x15C[NumBoostStates]).

- CoreFreq (MHz) = Calculated using the formula for CoreCOF.
- Power (mW) = MSRC001_00[6B:64][CpuVid] voltage * MSRC001_00[6B:64][IddValue] current * 1000.
- TransitionLatency (us) and BusMasterLatency (us):
 - If MSRC001_00[6B:64][CpuFid[5:0]] is the same for all enabled P-states (see MSRC001_00[6B:64][PstateEn]) and all boosted P-states:
 - TransitionLatency = BusMasterLatency = (15 steps * D0F0xBC_xD822_2114[GatersOffTime] time * 1000 us/ns) + (15 steps * D0F0xBC_xD822_2114[GatersOnTime] time * 1000 us/ns)
 - Else if MSRC001_00[6B:64][CpuFid[5:0]] is different for any enabled (see MSRC001_00[6B:64][PstateEn]) or boost P-states:
 - TransitionLatency = BusMasterLatency = (15 steps * D0F0xBC_xD822_2114[GatersOffTime] time * 1000 us/ns) + D0F0xBC_xD822_20B8[Pll_LOCK_TIMER] time + (15 steps * D0F0xBC_xD822_2114[GatersOnTime] time * 1000 us/ns)
 - Example:
 - MSRC001_00[6B:64][CpuFid[5:0]] is not the same for all P-states.
 - D0F0xBC_xD822_2114[GatersOffTime] = D0F0xBC_xD822_2114[GatersOnTime]= 5h (50 ns/step).
 - D0F0xBC_xD822_20B8[Pll_LOCK_TIMER] = 1b (2 us).
 - TransitionLatency = BusMasterLatency = (15 steps * 50 ns/step / 1000 us/ns) + 2 us + (15 steps * 50 ns/step / 1000 us/ns) = 3.5 us (round up to 4 us).
- Control/Status:
 - The highest performance non-boosted P-state must have the _PSS control and status fields programmed to 0.
 - Any lower performance non-boosted P-states must have the _PSS control and status fields programmed

in ascending order.

2.5.2.1.7.3.3 _PPC (Performance Present Capabilities)

The _PPC object is optional. Refer to the ACPI specification for details on use and content.

2.5.2.1.7.3.4 _PSD (P-state Dependency)

AMD recommends the ACPI 3.0 _PSD object be generated for each core as follows to cause the cores to transition between P-states independently:

- NumberOfEntries = 5.
- Revision = 0.
- Domain = CPUID Fn0000_0001_EBX[LocalApicId[7:2]].
- CoordType = FEh. (HW_ALL)
- NumProcessors = 2.

2.5.2.1.7.4 Fixed ACPI Description Table (FADT) Entries

Declare the following FADT entries:

- PSTATE_CNT = 00h.
- DUTY_WIDTH = 00h.

2.5.2.1.7.5 XPSS (Microsoft® Extended PSS) Object

Some Microsoft® operating systems require an XPSS object to make P-state changes function properly. A BIOS that implements an XPSS object has special requirements for the _PCT object. See the Microsoft *Extended PSS ACPI Method Specification* for the detailed requirements to implement these objects.

2.5.2.2 Core C-states

C-states are processor power states. C0 is the operational state in which instructions are executed. All other C-states are low-power states in which instructions are not executed. When coming out of warm and cold reset, the cores are transitioned to the C0 state.

2.5.2.2.1 C-state Names and Numbers

C-states are often referred to by an alphanumeric naming convention, C1, C2, C3, etc. The mapping between ACPI defined C-states and AMD specified C-states is not direct. AMD specified C-states are referred to as IO-based C-states. Up to three IO-based C-states are supported, IO-based C-state 0, 1, and 2. The IO-based C-state index corresponds to the offset added to MSRC001_0073[CstateAddr] to initiate a C-state request. See 2.5.2.2.2 [C-state Request Interface]. The actions taken by the processor when entering a low-power C-state are configured by software. See 2.5.2.2.3 [C-state Actions] for information about AMD specific actions.

2.5.2.2.2 C-state Request Interface

C-states are dynamically requested by software and are exposed through ACPI objects (see 2.5.2.2.6 [ACPI Processor C-state Objects]). C-states can be requested on a per-core basis. Software requests a C-state change in one of two ways:

- Reading from an IO address: The IO address must be the address specified by MSRC001_0073[CstateAddr]

plus an offset of 0 through 7. The processor always returns 0 for this IO read. Offsets 2 through 7 result in an offset of 2.
- Executing the HLT instruction. This is equivalent to reading from the IO address specified by D18F4x128[HaltCstateIndex].

When software requests a C-state transition, hardware evaluates any frequency and voltage domain dependencies and determines which C-state actions to execute. See 2.5.2.2.3 [C-state Actions].

2.5.2.2.3 C-state Actions

A core takes one of several different possible actions based upon a C-state change request from software. The C-state action fields are defined in D18F4x11[C:8].

2.5.2.2.3.1 C-state Probes and Cache Flushing

If probes occur after a core enters a non-C0 state, and the caches are not flushed by hardware, the core clock may be ramped back up to the C0 frequency to service the probes, as specified by D18F4x118/D18F4x11C[CpuPrbEn].

If a core enters a non-C0 state and cache flush is enabled (see D18F3xDC[CacheFlushOnHaltCtl] and D18F4x118/D18F4x11C[CacheFlushEn]), a timer counts down for a programmable period of time as specified by D18F3xDC[CacheFlushOnHaltTmr] or D18F4x118/D18F4x11C[CacheFlushTmrSel]. When the timer expires, the core flushes its L1 cache to L2 cache and the core clocks are ramped down to a divisor specified by D18F3xDC[CacheFlushOnHaltCtl]. The timer is reset if the core exits the C-state for any reason. See 2.5.2.2.4.2 [Cache Flush On Halt Saturation Counter].

Once a core flushes its caches, probes are no longer sent to that core. This improves probing performance for cores that are in C0.

2.5.2.2.3.2 Core C1 (CC1) State

When a core enters the CC1 state, its clock ramps down to the frequency specified by D18F4x118/D18F4x11C[ClkDivisorCstAct].

2.5.2.2.3.3 Core C6 (CC6) State

A core can gate off power to its internal logic when it enters any non-C0 state. This power gated state is known as CC6. In order to enter CC6, hardware first enters CC1 and then flushes the caches (see 2.5.2.2.3.1 [C-state Probes and Cache Flushing]) before checking D18F4x118/D18F4x11C[PwrGateEnCstAct]. Power gating reduces the amount of power consumed by the core.

The following sequence occurs when a core enters the CC6 state:

1. If MSRC001_0071[CurPstate] < D18F3xA8[PopDownPstate], transition the core P-state to D18F3xA8[PopDownPstate].
2. L1 and L2 caches are flushed to DRAM. See 2.5.2.2.3.1 [C-state Probes and Cache Flushing].
3. Internal core state is saved to DRAM.
4. Power is removed from the core.

All of the following must be true in order for a core to be placed into CC6:
- D18F4x118/D18F4x11C[CacheFlushEn] == 1 for the corresponding C-state action field.

- D18F4x118/D18F4x11C[CacheFlushTmrSel] != 11b for the corresponding C-state action field.
- D18F4x118/D18F4x11C[PwrGateEnCstAct] == 1 for the corresponding C-state action field.
- D18F2x118[CC6SaveEn] == 1.
- D18F2x118[LockDramCfg] == 1.
- The CC6 storage area in DRAM is configured. See 2.9.13 [DRAM CC6/PC6 Storage].

The events which cause a core to exit the CC6 state are specified in 2.5.2.2.5 [Exiting C-states].

If a warm reset occurs while a core is in CC6, all MCA registers in the core shown in Table 58 are cleared to 0. See 2.16.1 [Machine Check Architecture].

The time required to enter and exit CC6 is directly proportional to the core P-state frequency. Slower core frequencies require longer entry and exit times. Latency issues may occur with core P-state frequencies less than 800MHz.

2.5.2.2.3.4 Package C6 (PC6) State

When all cores enter a non-C0 state, VDD can be reduced to a non-operational voltage that does not retain core state. This state is known as PC6 and reduces the amount of static and dynamic power consumed by all cores. The following actions are taken by hardware prior to PC6 entry:

1. If MSRC001_0071[CurPstate] < D18F3xA8[PopDownPstate], transition the core P-state to D18F3xA8[PopDownPstate].
2. For all cores not in CC6, L1 and L2 caches are flushed to DRAM. See 2.5.2.2.3.1 [C-state Probes and Cache Flushing].
3. For all cores not in CC6, internal core state is saved to DRAM.
4. VDD is transitioned to the VID specified by D18F5x128[PC6Vid].

All of the following must be true on all cores in order for a package to be placed into PC6:

- D18F4x118/D18F4x11C[CacheFlushEn] == 1 for the corresponding C-state action field.
- D18F4x118/D18F4x11C[CacheFlushTmrSel] != 11b for the corresponding C-state action field.
- D18F4x118/D18F4x11C[PwrOffEnCstAct] == 1 for the corresponding C-state action field.
- D18F2x118[CC6SaveEn] == 1.
- D18F2x118[LockDramCfg] == 1.
- MSRC001_0015[HltXSpCycEn] == 1.

2.5.2.2.4 C-state Request Monitors

Deeper C-states have higher entry and exit latencies but provide greater power savings than shallower C-states. To help balance the performance and power needs of the system, the processor can limit access to specific C-states in certain scenarios.

2.5.2.2.4.1 FCH Messaging

The FCH can be notified when the processor transitions package C-states. See the following:

- D18F4x128[CstateMsgDis].
- D18F5x178[CstateFchDis].
- MSRC001_0015[HltXSpCycEn].

2.5.2.2.4.2 Cache Flush On Halt Saturation Counter

A cache flush success monitor tracks the success rate of cache flush timer expirations relative to the core exiting a C-state. Based on the success rate, caches may be flushed immediately without waiting for the cache flush timer to expire. See D18F4x128[CacheFlushSucMonThreshold]. When the core resumes normal execution, the caches refill as normal.

2.5.2.2.5 Exiting C-states

The following events may cause the processor to exit a non-C0 C-state and return to C0:
- INTR
- NMI
- SMI
- INIT
- RESET_L assertion

If an INTR is received while a core is in a non-C0 C-state, the state of EFLAGS[IF] and the mechanism used to enter the non-C0 C-state determine the actions taken by the processor.
- Entry via HLT, EFLAGS[IF] == 0: The interrupt does not wake up the core.
- Entry via HLT, EFLAGS[IF] == 1: The interrupt wakes the core and the core begins execution at the interrupt service routine.
- Entry via IO read, EFLAGS[IF] == 0: The interrupt wakes the core and the core begins execution at the instruction after the IN instruction that was used to enter the non-C0 C-state.
- Entry via IO read, EFLAGS[IF] == 1: The interrupt wakes the core and the core begins execution at the interrupt service routine.

2.5.2.2.6 ACPI Processor C-state Objects

Processor power control is implemented through the _CST object in ACPI 2.0 and later revisions. The presence of the _CST object indicates to the OS that the platform and processor are capable of supporting multiple power states. BIOS must provide the _CST object and define other ACPI parameters to support operating systems that provide native support for processor C-state transitions. See 2.5.2.2.6.1 [_CST]. See 2.5.2.2.6.2 [_CSD].

The _CST object is not supported with ACPI 1.0b. BIOS should provide FADT entries to support operating systems that are not compatible with ACPI 2.0 and later revisions. See 2.5.2.2.6.3 [_CRS].

2.5.2.2.6.1 _CST

The _CST object should be generated for each core as follows:
- Count = 1.
- Register = MSRC001_0073[CstateAddr] + 1.
- Type = 2.
- Latency = 400.
- Power = 0.

2.5.2.2.6.2 _CSD

The _CSD object should be generated for each core as follows:
- NumberOfEntries = 6.
- Revision = 0.
- Domain = CPUID Fn0000_0001_EBX[LocalApicId[7:1]].

- CoordType = FEh. (HW_ALL)
- NumProcessors = 2.
- Index = 0.

2.5.2.2.6.3 _CRS

BIOS must declare in the root host bridge _CRS object that the IO address range from MSRC001_0073[CstateAddr] to MSRC001_0073[CstateAddr]+7 is consumed by the host bridge.

2.5.2.2.6.4 Fixed ACPI Description Table (FADT) Entries

Declare the following FADT entries:

- P_LVL2_LAT = 100.
- P_LVL3_LAT = 1001.
- FLAGS.PROC_C1 = 1.
- FLAGS.P_LVL2_UP = 1.

Declare the following P_BLK entries:

- P_LVL2 = MSRC001_0073[CstateAddr] + 1.
- P_LVL3 = 0.

BIOS must declare the PSTATE_CNT entry as 00h.

2.5.2.2.7 BIOS Requirements for Initialization

1. Initialize MSRC001_0073[CstateAddr] with an available IO address. See 2.5.2.2.6.3 [_CRS].
2. Initialize D18F4x11[C:8].
3. Generate ACPI objects as described in 2.5.2.2.6 [ACPI Processor C-state Objects].

2.5.2.3 Effective Frequency

The effective frequency interface allows software to discern the average, or effective, frequency of a given core over a configurable window of time. This provides software a measure of actual performance rather than forcing software to assume the current frequency of the core is the frequency of the last P-state requested. This can be useful when the P-state is limited by:

- HTC
- D18F3x68[SwPstateLimit]
- SBI
- CPB

The following procedure calculates effective frequency using MSR0000_00E7 [Max Performance Frequency Clock Count (MPERF)] and MSR0000_00E8 [Actual Performance Frequency Clock Count (APERF)]:

1. At some point in time, write 0 to both MSRs.
2. At some later point in time, read both MSRs.
3. Effective frequency = (value read from MSR0000_00E8 / value read from MSR0000_00E7) * P0 frequency using software P-state numbering.

Additional notes:

- The amount of time that elapses between steps 1 and 2 is determined by software.
- It is software's responsibility to disable interrupts or any other events that may occur in between the write of MSR0000_00E7 and the write of MSR0000_00E8 in step 1 or between the read of MSR0000_00E7 and the

read of MSR0000_00E8 in step 2.
- The behavior of MSR0000_00E7 and MSR0000_00E8 may be modified by MSRC001_0015[EffFreqCntMwait].
- The effective frequency interface provides +/- 50MHz accuracy if the following constraints are met:
- Effective frequency is read at most one time per millisecond.
- When reading or writing MSR0000_00E7 and MSR0000_00E8 software executes only MOV instructions, and no more than 3 MOV instructions, between the two RDMSR or WRMSR instructions.
- MSR0000_00E7 and MSR0000_00E8 are invalid if an overflow occurs.

2.5.3 NB Power Management

2.5.3.1 NB P-states

The processor supports several P-states, labled NBP0 (highest frequency) through NBPn (where n is the number of the lowest frequency P-state). Four of the NB P-states are specified in D18F5x16[C:0]; the processor may be configured to support additional P-states that are not visible in registers as well.

Out of cold reset, the NB P-state is specified by D18F5x174[StartupNbPstate] and D0F0xBC_xD823_0F00[COF_VID_PROG]. The current NB P-state is specified by D18F5x174[CurNbFid, CurNbDid, CurNbVid].

Although several NB P-states are defined, only two NB P-states are used at any given time, specified by D18F5x170[NbPstateHi, NbPstateLo].

2.5.3.1.1 NB P-state Transitions

Hardware selects whether to use the high or low NB P-state. Once it is determined that an NB P-state transition is necessary, the NB executes the following sequence:

1. If transitioning from the low NB P-state to the high NB P-state, transition VDDNB voltage.
2. If the GPU is enabled as specified by D18F5x178[SwGfxDis], wait for the display buffer to fill.
3. Quiesce all active cores.
4. Stop memory traffic and place DRAM into self-refresh.
5. Transition NCLK frequency.
6. Update NB P-state specific DRAM settings within hardware.
7. Take DRAM out of self-refresh and allow memory traffic.
8. Wake up cores.
9. If transitioning from the high NB P-state to the low NB P-state, transition VDDNB voltage.

2.5.3.1.2 BIOS NB P-state Configuration

2.5.3.1.2.1 NB P-state COF and VID Synchronization After Warm Reset

BIOS performs the following sequence from one core. This is done after any warm reset and before 2.9.9 [DCT/DRAM Initialization and Resume].
1. Temp1 = D18F5x170[SwNbPstateLoDis].
2. Temp2 = D18F5x170[NbPstateDisOnP0].
3. Temp3 = D18F5x170[NbPstateThreshold].
4. Temp4 = D18F5x170[NbPstateGnbSlowDis].
5. If MSRC001_0070[NbPstate] == 0, go to step 6. If MSRC001_0070[NbPstate] == 1, go to step 11.
6. Write 1 to D18F5x170[NbPstateGnbSlowDis].
7. Write 0 to D18F5x170[SwNbPstateLoDis, NbPstateDisOnP0, NbPstateThreshold].

8. Waits for (D18F5x174[NbPstateReqBusy] == 0 && D18F5x174[CurNbPstateLo] == 1).
9. Set D18F5x170[SwNbPstateLoDis] = 1.
10. Waits for (D18F5x174[NbPstateReqBusy] == 0 && D18F5x174[CurNbPstateLo] == 0). Go to step 15.
11. Write 1 to D18F5x170[SwNbPstateLoDis].
12. Waits for (D18F5x174[NbPstateReqBusy] == 0 && D18F5x174[CurNbPstateLo] == 0).
13. Write 0 to D18F5x170[SwNbPstateLoDis, NbPstateDisOnP0, NbPstateThreshold].
14. Waits for (D18F5x174[NbPstateReqBusy] == 0 && D18F5x174[CurNbPstateLo] == 1).
15. Set D18F5x170[SwNbPstateLoDis] = Temp1, D18F5x170[NbPstateDisOnP0] = Temp2, D18F5x170[NbPstateThreshold] = Temp3 and D18F5x170[NbPstateGnbSlowDis] = Temp4.

2.5.3.1.2.2 NB P-state Transitions

During boot when D18F5x174[NbPstateDis] == 0, BIOS forces the processor to the desired NB P-states using the following steps:

1. Save the values in D18F5x170 for later restoration to unforce the NB P-state.
2. Force transitions if needed:
 a. If modifying NbPstateHi, force transition to NbPstateLo first (ensure D18F5x174[CurNbPstateLo] == 1 && D18F5x174[NbPstateReqBusy] == 0). To force, look at step 4.
 b. If modifying NbPstateLo, force transition to NbPstateHi first (ensure D18F5x174[CurNbPstateLo] == 0&& D18F5x174[NbPstateReqBusy] == 0). To Force, look at step 4.
3. Set the desired NB P-state pointers, D18F5x170[NbPstateHi, NbPstateLo].
4. Transition to the desired state as follows:
 - In order to transition to D18F5x170[NbPstateHi], program D18F5x170 as follows:
 - SwNbPstateLoDis = 1.
 - Wait for D18F5x174[CurNbPstateLo] == 0 && D18F5x174[NbPstateReqBusy] == 0.
 - In order to transition to D18F5x170[NbPstateLo], program D18F5x170 as follows:
 - SwNbPstateLoDis = NbPstateDisOnP0 = NbPstateThreshold = 0.
 - Wait for D18F5x174[CurNbPstateLo] == 1 && D18F5x174[NbPstateReqBusy] == 0.

BIOS performs the following to release the NB P-state force:

5. Release the NB P-state force by restoring initial D18F5x170 values.
 - Restore the initial D18F5x170[SwNbPstateLoDis, NbPstateDisOnP0, NbPstateLo] values.
 - Restore the initial D18F5x170[NbPstateThreshold, NbPstateHi] values.

2.5.3.2 NB C-states

NB C-states are package-level actions that occur only when all transactions through the NB have stopped. The NB C-state actions are:

- DRAM self-refresh (see 2.5.6.2 [DRAM Self-Refresh]):
 - Enable bit: D18F4x118/D18F4x11C[SelfRefr].
 - Entry requirements:
 - No outstanding GPU traffic or traffic from a link.
 - Exit conditions (any of the following must be true):
 - The local APIC timer expires. See 2.4.9.1 [Local APIC].
 - New GPU traffic or traffic from a link.
 - A P-state limit update (see 2.5.2.1.4 [Core P-state Limits]) causes the most restrictive P-state limit to become a higher number than the current P-state for any core in CC1.

- NB clock gating:

- Enable bit: D18F4x118/D18F4x11C[NbClkGate].
- Entry requirements:
 - No outstanding GPU traffic or traffic from a link.
- Exit conditions (any of the following must be true):
 - The local APIC timer expires. See 2.4.9.1 [Local APIC].
 - New GPU traffic or traffic from a link.
 - A P-state limit update (see 2.5.2.1.4 [Core P-state Limits]) causes the most restrictive P-state limit to become a higher number than the current P-state for any core in CC1.

NOTE: For SERVER system solutions, NB power gating should be disabled by setting D18F4x118/D18F4x11C[NbPwrGate] = 0 in the BIOS. BIOS setting of D18F4x118/D18F4x11C[NbPwrGate] = 1 should be set for client systems.

- NB power gating:
 - Enable bit: D18F4x118/D18F4x11C[NbPwrGate].
 - Entry requirements (all of the following must be true):
 - No outstanding GPU traffic or traffic from a link.
 - All cores are in CC6.
 - DRAM is either in or entering self-refresh.
 - Exit conditions (any of the following must be true):
 - The local APIC timer expires. See 2.4.9.1 [Local APIC].
 - New GPU traffic or traffic from a link.
 - A P-state limit update (see 2.5.2.1.4 [Core P-state Limits]) causes the most restrictive P-state limit to become a higher number than the current P-state for any core in CC1.

When entering NB C-states, the actions are taken in the following order:
1. DRAM self-refresh.
2. NB clock gating.
3. NB power gating.

When exiting NB C-states, the actions are taken in the following order:
1. NB power gating.
2. NB clock gating.
3. DRAM self-refresh.

2.5.4 Bandwidth Requirements

- The frequency relationship of (core COF / NB COF) <= 6 must be maintained for all supported P-state combinations (e.g., a core P0 COF of 4.0 GHz could not be combined with a NB P0 COF of 0.6 GHz) the NB P0 COF would have to be 0.8 GHz or greater; if the NB P0 COF is 1.2 GHz, then the NB P1 COF of 0.6 GHz may only be supported if the corresponding core P-state specify a COF of 3.0 GHz or less.
- All core P-states are required to be defined such that (NB COF/core COF) <= 32, for all NB/core P-state combinations (e.g., if the NB COF is 4.8 GHz then the core COF must be no less than 150 MHz).
- All core P-states must be defined such that CoreCOF >= 500 MHz.
- All core P-states must be defined such that MSRC001_00[6B:64][CpuFid[5:0]] <= 22h.
- All NB P-states must be defined such that D18F5x16[C:0][NbFid] <= 2Eh.
- NBCOF >= MEMCLK frequency.

2.5.5 GPU Power Management

The processor supports dynamic GPU frequency changes along with VDDNB voltage change requests, known as Dynamic Power Management (DPM). Once initialized, hardware dynamically monitors processor

utilization and adjusts the frequencies and voltage based on that utilization. For DPM, higher numbered states represent higher performance and lower numbered states represent lower performance.

2.5.6 DRAM Power Management

2.5.6.1 Memory P-states

The processor supports up to 2 memory P-states, M0 and M1. Each memory P-state consists of the following:

- MEMCLK frequency.
- A set of frequency dependent DRAM timing and configuration registers.

See 2.9 [DRAM Controllers (DCTs)] for DRAM technology specific information and requirements.

All valid memory P-states are associated with a specific NB P-state. When hardware transitions to a new NB P-state, the memory P-state is transitioned to that specified by the new NB P-state.

Out of cold reset the current memory P-state is M0. The P-state value specified by D18F5x16[C:0][MemPstate] of the NB P-state indexed by D18F5x174[StartupNbPstate] is invalid. Support for dynamic memory P-state changes is indicated by D18F3xE8[MemPstateCap] == 1 and one or more D18F5x16[C:0][MemPstate] == 1; otherwise M0 is used by hardware for configuration purposes.

During boot, and if D18F5x170[MemPstateDis]=0, the BIOS can disable memory P-states using the following steps:

1. Program D18F5x170[MemPstateDis] = 1.
2. Program D18F5x16[C:0][MemPstate] = 0.

2.5.6.2 DRAM Self-Refresh

DRAM is placed into self-refresh on S3 entry. In addition to S3, DRAM is placed into self-refresh in S0 in the following two scenarios:

- NB P-state transitions (see 2.5.3.1 [NB P-states]).
- NB C-states (see 2.5.3.2 [NB C-states]).

The following requirements must be met before hardware places DRAM into self-refresh:

- No pending traffic.
- One of the following is true:
 - The GPU is idle and the internal display buffer is full.
 - The internal GPU is disabled.

Once the above requirements are met, hardware places DRAM into self-refresh.

Early self-refresh occurs when DRAMs are placed in self-refresh before expiration of the cache flush timer. See D18F4x118/D18F4x11C[SelfRefrEarly] and D18F5x128[SelfRefrEarlyDis]. If early self-refresh is enabled, the DRAMs are taken out of self-refresh to perform the flush operation when the cache flush timer expires and then placed back into self-refresh.

To save additional power, hardware always tristates MEMCLK when entering self-refresh.

2.5.6.3 Stutter Mode

DRAM is most commonly placed in self-refresh due to stutter mode when the internal GPU is in use. The display buffer in the GPU is a combination of a large buffer known as the DMIF (Display Memory Interface FIFO) and a smaller line buffer. The DMIF takes data originating from DRAM and sends it to the line buffer to draw to the screen. When the data level in the DMIF is full, DRAM is placed in self-refresh, and incoming DRAM requests are queued. As the DMIF drains, it eventually falls below a predefined watermark level, at which point hardware pulls DRAM out of self-refresh and services all the requests in the queue. Once all the requests are complete and the DMIF is full again, a transition back into self-refresh occurs if the stutter mode conditions are still met.

2.5.6.4 EVENT_L

EVENT_L is a level sensitive input to the processor. When asserted, the actions specified by D18F2xA4 are taken. EVENT_L is generally asserted to indicate that a DRAM high temperature condition exists. The minimum assertion time for EVENT_L is 15 ns. The minimum deassertion time for EVENT_L is 15 ns.

- EVENT_L is pulled to VDDIO on the motherboard.
- EVENT_L is ignored while:
 - PWROK is de-asserted.
 - RESET_L is asserted.
- BIOS must ensure that throttling is disabled (see D18F2xA4[CmdThrottleMode]) until DRAM training is complete.

See 2.9.14 [DRAM On DIMM Thermal Management and Power Capping].

2.5.7 System Power Management States

2.5.7.1 S-states

S-states are ACPI defined sleep states. S0 is the operational state. All other S-states are low-power states in which various voltage rails in the system may or may not be powered. See the ACPI specification for descriptions of each S-state.

2.5.7.1.1 ACPI Suspend to RAM State (S3)

The processor supports the ACPI-defined S3 state. Software is responsible for restoring the state of the processor's registers when resuming from S3. All registers in the processor that BIOS initialized during the initial boot must be restored. The method used to restore the registers is system specific.

During S3 entry, software is responsible for transitioning the processor to Memory Pstate0. See 2.5.6.1 [Memory P-states].

During S3 entry, system memory enters self-refresh mode (see 2.5.6.2 [DRAM Self-Refresh]). Software is responsible for bringing memory out of self-refresh mode when resuming from S3.

Many of the systemboard power planes for the processor are powered down during S3. Refer to the Electrical Data Sheet.

2.5.8 Application Power Management (APM)

Application Power Management (APM) allows the processor to provide maximum performance while remaining within the specified power delivery and removal envelope. APM dynamically monitors processor

activity and generates an approximation of power consumption. If power consumption exceeds a defined power limit, a P-state limit is applied by APM hardware to reduce power consumption. APM ensures that average power consumption over a thermally significant time period remains at or below the defined power limit. This allows P-states to be defined with higher frequencies and voltages than could be used without APM.

2.5.8.1 Core Performance Boost (CPB)

These P-states are referred to as boosted P-states.

- Support for APM is specified by CPUID Fn8000_0007_EDX[CPB].
- APM is enabled if all of the following conditions are true:
 - MSRC001_0015[CpbDis] == 0 for all cores.
 - D18F4x15C[ApmMasterEn] == 1.
 - D18F4x15C[BoostSrc] == 01b.
 - D18F4x15C[NumBoostStates] != 0.
- APM can be dynamically enabled and disabled through MSRC001_0015[CpbDis]. If core performance boost (CPB) is disabled on any core, a P-state limit is applied to all cores. The P-state limit restricts cores to the highest performance non-boosted P-state.
- All P-states, both boosted and non-boosted, are specified in MSRC001_00[6B:64].
- The number of boosted P-states is specified by D18F4x15C[NumBoostStates].
 - The number of boosted P-states may vary from product to product.
- Two types of boosted P-states are supported. Cores can be placed in the first type of boosted P-states if the processor current consumption remains within the electrical limit with all cores active. The second type of boosted P-states is C-state Boost. See 2.5.8.1.1 [C-state Boost].
- All boosted P-states are always higher performance than non-boosted P-states.
- To ensure proper operation, boosted P-states are from the operating system. BIOS does not provide ACPI _PSS entries for boosted P-states. See 2.5.2.1.7.3.2 [_PSS (Performance Supported States)].
- The lowest-performance P-state CPB limits the processor to is the highest-performance non-boosted P-state.

2.5.8.1.1 C-state Boost

C-state Boost can only be achieved if a subset of cores/compute units are in CC6 and the processor current consumption remains within the electrical limit. See D18F4x16C[CstateCnt, CstateBoost, CstateCores].

2.5.8.2 Thermal Limiting

Thermal limiting is a mechanism for capping the power consumption of the processor through a thermal limit.

2.5.8.3 Bidirectional Application Power Management (BAPM)

Bidirectional Application Power Management (BAPM) is an algoirthm to enable fine grained power transfers between the core and GPU.

2.6 Performance Monitoring

The processor includes support for two methods of monitoring processor performance:
- 2.6.1 [Performance Monitor Counters].
- 2.6.2 [Instruction Based Sampling (IBS)].

2.6.1 Performance Monitor Counters

The following types of performance counters are supported:
- 2.6.1.1 [Core Performance Monitor Counters], consisting of one set located in each core of each compute unit.
- 2.6.1.2 [NB Performance Monitor Counters], consisting of one set located in each node.

The accuracy of the performance counters is not ensured. The performance counters are not assured of producing identical measurements each time they are used to measure a particular instruction sequence, and they should not be used to take measurements of very small instruction sequences. The RDPMC instruction is not serializing, and it can be executed out-of-order with respect to other instructions around it. Even when bound by serializing instructions, the system environment at the time the instruction is executed can cause events to be counted before the counter value is loaded into EDX:EAX.

To accurately start counting with the write that enables the counter, disable the counter when changing the event and then enable the counter with a second MSR write.

Writing the performance counters can be useful if there is an intention for software to count a specific number of events, and then trigger an interrupt when that count is reached. An interrupt can be triggered when a performance counter overflows. Software should use the WRMSR instruction to load the count as a two's-complement negative number into the performance counter. This causes the counter to overflow after counting the appropriate number of times.

In addition to the RDMSR instruction, the performance counter registers can be read using a special read performance-monitoring counter instruction, RDPMC.

2.6.1.1 Core Performance Monitor Counters

The core performance monitor counters are used by software to count specific events that occur in a core of the compute unit. Each core of each compute unit provides six 48-bit performance counters. Unless otherwise specified, the events count only the activity of the core, not activity caused by the other core of the compute unit.

MSRC001_020[A,8,6,4,2,0] [Performance Event Select (PERF_CTL[5:0])] specify the events to be monitored and how they are monitored. MSRC001_020[B,9,7,5,3,1] [Performance Event Counter (PERF_CTR[5:0])] are the counters. MSRC001_00[03:00] is the legacy alias for MSRC001_020[6,4,2,0]. MSRC001_00[07:04] is the legacy alias for MSRC001_020[7,5,3,1]. Support for MSRC001_020[B:0] is indicated by CPUID Fn8000_0001_ECX[PerfCtrExtCore].

All of the events are specified in 3.24 [Core Performance Counter Events].

Some performance monitor events have a maximum count per clock that exceeds one event per clock. These performance events are called multi-events. Some counters support a greater multi-event count per clock than others. Events that are multi-events specify the maximum multi-event count per clock (e.g., The number of events logged per cycle can vary from 0 to X). An event that doesn't specify multi-event is implied to be a

maximum of 1 event per clock. Undefined results are produced if an multi-event is selected that exceeds that counters capabilities. The following list specifies the maximum number of multi-events supported by each counter:

- PERF_CTL[0]: 31 multi-event per clock maximum.
- PERF_CTL[1]: 7 multi-event per clock maximum.
- PERF_CTL[2]: 7 multi-event per clock maximum.
- PERF_CTL[3]: 63 multi-event per clock maximum.
- PERF_CTL[4]: 7 multi-event per clock maximum.
- PERF_CTL[5]: 7 multi-event per clock maximum.

Not all performance monitor events can be counted on all counters. The performance counter registers are generally assigned to specific blocks of the core according to Table 11; however, there are exceptions when an event is implemented by another block of the core and therefore has the counter restrictions of that block. Each core event description starts with one of the following terms to indicate which counters support that event. Selecting an event for a counter that does not support that counter produces undefined results.

Table 11: Core PMC mapping to PERF_CTL[5:0]

Term	Definition
PERF_CTL[5:0]	PERF_CTL[5:0] are used to count events in the LS/DC and EX where the number of events logged per cycle can vary up to 7.
PERF_CTL[3,0]	PERF_CTL[3,0] are used to count events in the LS/DC and EX where the number of events logged per cycle can vary up to 31.
PERF_CTL[0]	PERF_CTL[0] are used to count events in the LS/DC, EX, IF/DE and CU where the number of events logged per cycle can vary up to 31.
PERF_CTL[3]	PERF_CTL[3] are used to count events in the LS/DC, EX and FP where the number of events logged per cycle can vary up to 63.
PERF_CTL[2:0]	PERF_CTL[2:0] are used to count events in the IF/DE and CU; The number of events logged per cycle can vary up to 7.
PERF_CTL[5:3]	PERF_CTL[5:3] are used to count events in the FP; The number of events logged per cycle can vary up to 7.

2.6.1.2 NB Performance Monitor Counters

The NB performance monitor counters are used by software to count specific events that occur in the NB. Each node provides four 48-bit performance counters. Since the northbridge performance counter register are shared by all cores on a node, monitoring of northbridge events should only be performed by one core on a node.

These counters are 48 bits. This requires two 32-bit reads to get the entire value. The high bits are not latched when the low bits are read. This means that it is possible for the low bits to overflow into the high bits between the two reads. Software can read the registers multiple times in the sequence below to ensure the proper value is read.

1. Read the low bits[31:0], call the result Lo0.
2. Read the high bits[47:32], call the result Hi0.
3. Read the low bits again, call the result Lo1.
4. Read the high bits again, call the result Hi1.
5. If Hi1 == Hi0, a roll over did not occur and {Hi0,Lo0} is valid.
6. If Hi1 != Hi0, a roll over may have occurred. If so, read again to get Hi2 and Lo2.
7. If Hi2 == Hi1, then {Hi1,Lo1} is a valid counter value.

MSRC001_024[6,4,2,0] [Northbridge Performance Event Select (NB_PERF_CTL[3:0])] and

MSRC001_024[7,5,3,1] [Northbridge Performance Event Counter (NB_PERF_CTR[3:0])] specify the events to be monitored and how they are monitored. Support for MSRC001_024[7:0] is indicated by CPUID Fn8000_0001_ECX[PerfCtrExtNB].

All of the events are specified in 3.25 [NB Performance Counter Events].

All NB performance monitor events can be counted on all counters.

All NB performance events are one event per clock.

NB performance counters do not support APIC interrupt capability.

2.6.2 Instruction Based Sampling (IBS)

IBS is a code profiling mechanism that enables the processor to select a random instruction fetch or micro-op after a programmed time interval has expired and record specific performance information about the operation. An interrupt is generated when the operation is complete as specified by MSRC001_103A [IBS Control]. An interrupt handler can then read the performance information that was logged for the operation.

The IBS mechanism is split into two parts: instruction fetch performance controlled by MSRC001_1030 [IBS Fetch Control (IbsFetchCtl)]; and instruction execution performance controlled by MSRC001_1033 [IBS Execution Control (IbsOpCtl)]. Instruction fetch sampling provides information about instruction TLB and instruction cache behavior for fetched instructions. Instruction execution sampling provides information about micro-op execution behavior. The data collected for instruction fetch performance is independent from the data collected for instruction execution performance. Support for the IBS feature is indicated by the CPUID Fn8000_0001_ECX[IBS].

Instruction fetch performance is profiled by recording the following performance information for the tagged instruction fetch:

- If the instruction fetch completed or was aborted. See MSRC001_1030.
- The number of clock cycles spent on the instruction fetch. See MSRC001_1030.
- If the instruction fetch hit or missed the IC, hit/missed in the L1 and L2 TLBs, and page size. See MSRC001_1030.
- The linear address, physical address associated with the fetch. See MSRC001_1031, MSRC001_1032.

Instruction execution performance is profiled by tagging one micro-op associated with an instruction. Instructions that decode to more than one micro-op return different performance data depending upon which micro-op associated with the instruction is tagged. These micro-ops are associated with the RIP of the next instruction to retire. The following performance information is returned for the tagged micro-op:

- Branch and execution status for micro-ops. See MSRC001_1035.
- Branch target address for branch micro-ops. See MSRC001_103B.
- The logical address associated with the micro-op. See MSRC001_1034.
- The linear and physical address associated with a load or store micro-op. See MSRC001_1038, MSRC001_1039.
- The data cache access status associated with the micro-op: DC hit/miss, DC miss latency, TLB hit/miss, TLB page size. See MSRC001_1037.
- The number clocks from when the micro-op was tagged until the micro-op retires. See MSRC001_1035.
- The number clocks from when the micro-op completes execution until the micro-op retires. See MSRC001_1035.
- Source information for DRAM and MMIO. See MSRC001_1036.

2.7 Configuration Space

PCI-defined configuration space was originally defined to allow up to 256 bytes of register space for each function of each device; these first 256 bytes are called base configuration space (BCS). It was expanded to support up to 4096 bytes per function; bytes 256 through 4095 are called extended configuration space (ECS). The processor includes configuration space registers located in both BCS and ECS. Processor configuration space is accessed through bus 0, devices 18h to 1Fh, where device 18h corresponds to node 0 and device 1Fh corresponds to node 7. See 2.7.3 [Processor Configuration Space].

Configuration space is accessed by the processor through two methods as follows:
- IO-space configuration: IO instructions to addresses CF8h and CFCh.
 - Enabled through IOCF8[ConfigEn], which allows access to BCS.
 - Access to ECS enabled through MSRC001_001F [EnableCf8ExtCfg].
 - Use of IO-space configuration can be programmed to generate GP faults through MSRC001_0015[IoCfgGpFault].
 - SMI trapping for these accesses is specified by MSRC001_0054 [IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)] and MSRC001_00[53:50] [IO Trap (SMI_ON_IO_TRAP_[3:0])].
- MMIO configuration: configuration space is a region of memory space.
 - The base address and size of this range is specified by MSRC001_0058 [MMIO Configuration Base Address]. The size is controlled by the number of configuration-space bus numbers supported by the system. Accesses to this range are converted configuration space as follows:
 - Address[31:0] = {0h, bus[7:0], device[4:0], function[2:0], offset[11:0]}.

The BIOS may use either configuration space access mechanism during boot. Before booting the OS, BIOS must disable IO access to ECS, enable MMIO configuration and build an ACPI defined MCFG table. BIOS ACPI code must use MMIO to access configuration space.

Per the link specification, BCS accesses utilize link addresses starting at FD_FE00_0000h and ECS accesses utilize link addresses starting at FE_0000_0000h.

2.7.1 MMIO Configuration Coding Requirements

MMIO configuration space accesses must use the uncacheable (UC) memory type.
Instructions used to read MMIO configuration space are required to take the following form:

```
mov eax/ax/al, any_address_mode;
```

Instructions used to write MMIO configuration space are required to take the following form:

```
mov any_address_mode, eax/ax/al;
```

No other source/target registers may be used other than eax/ax/al.

In addition, all such accesses are required not to cross any naturally aligned DW boundary. Access to MMIO configuration space registers that do not meet these requirements result in undefined behavior.

2.7.2 MMIO Configuration Ordering

Since MMIO configuration cycles are not serializing in the way that IO configuration cycles are, their ordering rules relative to posted may result in unexpected behavior.

Therefore, processor MMIO configuration space is designed to match the following ordering relationship that exists naturally with IO-space configuration: if a core generates a configuration cycle followed by a posted-write cycle, then the posted write is held in the processor until the configuration cycle completes. As a result, any unexpected behavior that might have resulted if the posted-write cycle were to pass MMIO configuration cycle is avoided.

2.7.3 Processor Configuration Space

The processor includes configuration space as described in 3 [Registers]. Accesses to unimplemented registers of implemented functions are ignored: writes dropped; reads return 0. Accesses to unimplemented functions also ignored: writes are dropped; however, reads return all F's. The processor does not log any master abort events for accesses to unimplemented registers or functions.

Accesses to device numbers of devices not implemented in the processor are routed based on the configuration map registers. If such requests are master aborted, then the processor can log the event.

2.8 Northbridge (NB)

Each node includes a single northbridge that provides the interface to the local core(s), the interface to system memory, and the interface to system IO devices. The NB includes all power planes except VDD; see 2.5.1 [Processor Power Planes And Voltage Control].

The NB is responsible for routing transactions sourced from cores and link to the appropriate core, cache, DRAM, or link. See 2.4.6 [System Address Map].

2.8.1 NB Architecture

Major NB blocks are: System Request Interface (SRI), Memory Controller (MCT), DRAM Controllers (DCTs), and crossbar (XBAR). SRI interfaces with the core(s). MCT maintains cache coherency and interfaces with the DCTs; MCT maintains a queue of incoming requests called MCQ. XBAR is a switch that routes packets between SRI, MCT, and the link.

The MCT operates on physical addresses. Before passing transactions to the DCTs, the MCT converts physical addresses into *normalized* addresses that correspond to the values programmed into D18F2x[5C:40]_dct[0] [DRAM CS Base Address]. Normalized addresses include only address bits within the DCTs' range.

2.8.2 NB Routing

2.8.2.1 Address Space Routing

There are four main types of address space routed by the NB:
1. Memory space targeting system DRAM.
2. Memory space targeting IO (MMIO).
3. IO space.
4. Configuration space.

2.8.2.1.1 DRAM and MMIO Memory Space

Memory space transactions provide the NB with the physical address, cacheability type, access type, and DRAM/MMIO destination type as specified in section 2.4.6.1.2 [Determining The Access Destination for Core Accesses].
Memory-space transactions are handled by the NB as follows:

- IO-device accesses are compared against:
 - If the access matches D18F1x[2CC:2A0,1CC:180,BC:80] [MMIO Base/Limit], then the transaction is routed to the root complex;
 - Else, if the access matches D18F1x[17C:140,7C:40] [DRAM Base/Limit], then the access is routed to the DCT;
 - Else, the access is routed to the UMI.
- For core accesses the routing is determined based on the DRAM/MMIO destination:
 - If the destination is DRAM:
 - If the access matches D18F1x[17C:140,7C:40] [DRAM Base/Limit], then the transaction is routed to the DCT;
 - Else, the access is routed to the UMI.
 - If the destination is MMIO:
 - If the access matches the VGA-compatible MMIO address space and D18F1xF4[VE] == 1 then D18F1xF4 describes how the access is routed and controlled;
 - Else, If the access matches D18F1x[2CC:2A0,1CC:180,BC:80] [MMIO Base/Limit], then the transaction is routed to the root complex;
 - Else, the access is routed to the UMI.

2.8.2.1.2 IO Space

IO-space transactions from IO links or cores are routed as follows:
- If the access matches D18F1x[DC:C0] [IO-Space Base/Limit], then the transaction is routed to the root complex;
- Else, If the access matches the VGA-compatible IO address space and D18F1xF4[VE] == 1 then D18F1xF4 describes how the access is routed and controlled.
- Else, the access is routed to the UMI.

2.8.2.1.3 Configuration Space

Configuration-space transactions from IO links are master aborted. Configuration-space transactions from cores are routed as follows:
- If the access matches D18F1x[1DC:1D0,EC:E0] [Configuration Map], then the transaction is routed to the specified link;
- Else, the access is routed to link that contains compatibility (subtractive) address space.

2.8.2.1.3.1 Recommended Buffer Count Settings Overview

When changing from the recommended settings, see the register programming requirements in the definition of each register. Some chipsets may further optimize these settings for their platform. If values other than the recommended settings are used, see the register requirements in the definition of each register. Table 12 defines commonly used terms for the following tables.

Table 12: ONION Link Definitions

Term	Definition
LinkGanged	Ganged = 0.
IOMMU	Indicates the presence of an IOMMU device on the IOH. IOMMU uses the isochronous flow control channel. If an IOMMU is present, D18F0x[E4,C4,A4,84][IsocEn] must be set for all links.
IFCM	Isochronous Flow Control Mode. IFCM = D18F0x[E4,C4,A4,84][IsocEn].

2.9 DRAM Controllers (DCTs)

The processor includes one DRAM controller (DCT). Each DCT controls one 64-bit DDR3 or DDR4 DRAM channel.

A DRAM channel consists of the group of DRAM interface pins connecting to one series of DIMMs.
BIOS reads D18F5x84[DctEn] to determine the DCT to DDR channel mapping as follows:
- DCT0 controls channel B when using the FP4 or AM4 package, and the lone memory channel available in the FT4 package. See CPUID Fn8000_0001_EBX[PkgType]

The DCTs operate on normalized addresses corresponding to the values programmed into D18F2x[5C:40]_dct[0]. Normalized addresses only include address bits within a DCT's range. The physical to normalized address translation varies based on various Northbridge settings. See 2.9.12 [Memory Hoisting] and 2.9.13 [DRAM CC6/PC6 Storage].

The following restrictions limit the DIMM types and configurations supported by the DCT:
- All DIMMs connected to a node are required to operate at the same MEMCLK frequency, regardless of the channel. All DCTs of a node must be programmed to the same frequency.
- ECC DIMMs are not supported.
- RDIMM, LRDIMM, and stacked DRAM type DIMMs are not supported.
- Mixing DR x16 8-bank DDR4 DIMMs with Address Mirroring with other DR 16-bank DIMMs with Address Mirroring are not supported.

2.9.1 Common DCT Definitions

Table 13: DCT Definitions

Term	Definition
AutoSelfRefresh	DDR3 SPDByte[31][2] of the DIMM being configured.
DataMaskMbType	Motherboard type for processor Data Mask pins. Bits — Description 00b — No connect 01b — Pins are routed per DM rules 10b — Pins are routed per DQS rules
Ddr3Mode	The DRAM controller and the phy are configured for DDR3 mode. See D18F2x78_dct[0][DramType].
Ddr4Mode	The DRAM controller and the phy are configured for DDR4 mode. See D18F2x78_dct[0][DramType].
DdrRate	The DDR data rate (MT/s) as specified by D18F2x94_dct[0][MemClkFreq] and D18F2x2E0_dct[0][M1MemClkFreq].
DeviceWidth	DDR3 SPDByte[7][2:0] of the DIMM being configured. DDR4 SPDByte[12][2:0] of the DIMM being configured.
DIMM	The DIMM being configured.
DIMM0	DIMM slots 0-n. The DIMMs on each channel are numbered from 0 to n where DIMM0 is the DIMM closest to the processor on that channel and DIMMn is the DIMM farthest from the processor on that channel.
DIMM1	

Table 13: DCT Definitions (Continued)

Term	Definition
DimmsPopulated	The number of DIMMs populated per channel plus rows of Solder-down DRAM devices.
DR	Dual Rank.
DramCapacity	DDR3 SPDByte[4][3:0] of the DIMM being configured. DDR4 SPDByte[4][3:0] of the DIMM being configured.
ExtendedTemperature-Range	DDR3 SPDByte[31][0] of the DIMM being configured.
MRS	JEDEC defined DRAM Mode Register Set.
NP	No DIMM populated.
NumDimmSlots	The number of motherboard DIMM slots per channel plus rows of Solder-down DRAM devices.
NumRanks	DDR3 SPDByte[7][5:3] of the DIMM being configured, or the number of ranks soldered down. DDR4 SPDByte[6].
Rank	The rank being configured.
RankMap	DDR3 SPDByte[63][0] of the DIMM being configured. DDR4 SPDByte[131][0] (unbuffered) or SPDByte[136][0] (registered or load reduced).
RowAddrBits	DDR3 SPDByte[5][5:3] of the DIMM being configured. DDR4 SPDByte[5][5:3] of the DIMM being configured.
Solder-down DRAM	DRAM devices soldered directly to the motherboard.
SODIMM	DCT is configured for SODIMM if (D18F2x90_dct[0][UnbuffDimm] == 1) and SODIMMs are populated.
SPD	Serial Presence Detect. In the case of DRAMs soldered on the platform, this refers to a virtual representation of the DRAM vendors' data sheets.
SR	Single Rank.
UDIMM	DCT is configured for UDIMM if (D18F2x90_dct[0][UnbuffDimm] == 1) and UDIMMs are populated.
VDDIO	DDR VDDIO in V.

2.9.2 DCT Frequency Support

The tables below list the maximum DIMM speeds supported by the processor for different configurations. The motherboard should comply with the relevant AMD socket motherboard design guideline (MBDG) to achieve the rated speeds. In cases where MBDG design options exist, lower-quality options may compromise the maximum achievable speed; motherboard designers should assess the tradeoffs.

The following are the DCT operational clocking requirements:
- NCLK >= MEMCLK frequency

Processors may support two different voltage levels on the VDDP rail based on MaxDdrRate. At the 1.05V nominal setting the maximum DDR rate is supported, while at the 0.95V setting the maximum DDR rate sup-

ported is DDR-2133.

Table 14: DDR3 UDIMM Maximum Frequency Support FP4/FT4 package

Num-DimmS-lots	Dimms Popu-lated	DRAM		Frequency (MT/s)		
		SR	DR	1.5V	1.35V	1.25V
1	1	1	-	1866	1866	-
		-	1	1866	1866	-
2	1	1	-	1866	1866	-
		-	1	1866	1866	-
	2	2	-	1866	1600	-
		1	1	1866	1600	-
		-	2	1866	1600	-

Table 15: DDR3 SODIMM Maximum Frequency Support FP4/FT4 package

Num-DimmS-lots	Dimms Popu-lated	DRAM		Frequency (MT/s)		
		SR	DR	1.5V	1.35V	1.25V
1	1	1	-	1866	1866	-
		-	1	1866	1866	-
2	1	1	-	1600	1600	-
		-	1	1600	1600	-
	2	2	-	1600	1600	-
		1	1	1600	1333	-
		-	2	1600	1333	-

Table 16: DDR4 UDIMM Maximum Frequency Support

Num-DimmS-lots	Dimms Popu-lated	DRAM		Frequency (MT/s)
		SR	DR	1.2V
1	1	1	-	2400
		-	1	2400
2	1	1	-	2400
		-	1	2133
	2	2	-	1866
		1	1	1866
		-	2	1866

Table 17: DDR4 SODIMM Maximum Frequency Support FP4 package

Num-DimmS-lots	Dimms Popu-lated	DRAM		Frequency (MT/s)
		SR	DR	1.2V
1	1	1	-	2133
		-	1	2133
2	1	1	-	1866
		-	1	1866
	2	2	-	1866
		1	1	1866
		-	2	1866

Table 18: DDR4 SODIMM Maximum Frequency Support FT4 package

Num-DimmS-lots	Dimms Popu-lated	DRAM		Frequency (MT/s)
		SR	DR	1.2V
1	1	1	-	2400
		-	1	2400
2	1	1	-	1866
		-	1	1866
	2	2	-	1866
		1	1	1866
		-	2	1866

The table below lists the DIMM populations as supported by the processor. DIMMs must be populated from farthest slot to closest slot to the processor on a per channel basis when a data bus daisy chain topology is used. DIMMs may be populated in either slot on a per channel basis when a data bus tee route topology is used.

Table 19: DDR Population Support Balanced Tee Route

NumDimmSlots	DIMM0	DIMM1
1	SR/DR	N/A
2	NP	SR/DR
	SR/DR	NP
	SR/DR	SR/DR

Table 20: DDR Population Support Daisy Chain or Unbalanced Tee Route

NumDimmSlots	DIMM0	DIMM1
1	SR/DR	N/A
2	NP	SR/DR
	SR/DR	SR/DR

2.9.3 DCT Configuration Registers

There are multiple types of DCT configuration registers:

- Registers for which there is one instance (e.g., D18F2xA4 or D18F2x78_dct[0]).
- Registers for which there is one instance per memory P-state for software accesses IOCF8[ExtRegNo[11]]. The syntax for this register type is described by example as follows:
 - D18F2x200_dct[0]_mp[1:0] refers to all instances of the D18F2x200 (M0) and D18F2xA00 (M1) registers.
 - D18F2x200_dct[0]_mp[1] refers to the register for memory P-state 1 (at D18F2xA00) of either or both DCTs.

2.9.4 DDR Pad to Processor Pin Mapping

The relationship of pad drivers to processor pins varies by package as shown in the following table.

Table 21: Package Pin Mapping DDR3/DDR4

Pad	Pin[1]
	FP4
MEMCLK0_H[0]	MA_CLK_H[0]
MEMCLK0_H[1]	MA_CLK_H[1]
MEMCLK0_H[2]	MA_CLK_H[2]
MEMCLK0_H[3]	MA_CLK_H[3]
MEMCS0_L[0]	MA0_CS_L[0]
MEMCS0_L[1]	MA0_CS_L[1]
MEMCS0_L[2]	MA1_CS_L[0]
MEMCS0_L[3]	MA1_CS_L[1]
MEMCS0_L[4]	NC
MEMCS0_L[5]	NC
MEMCS0_L[6]	NC
MEMCS0_L[7]	NC
MEMODT0[0]	MA0_ODT[0]
MEMODT0[1]	MA1_ODT[0]
MEMODT0[2]	MA0_ODT[1]
MEMODT0[3]	MA1_ODT[1]
MEMCKE0[0]	MA_CKE[0]
MEMCKE0[1]	MA_CKE[1]

Table 21: Package Pin Mapping DDR3/DDR4 (Continued)

MEMCKE0[2]	NC
MEMCKE0[3]	NC

1. For differential pins, only positive polarity pins are shown; negative polarity pins have corresponding mapping and are controlled by the same CSR field. NC = Not connected. BIOS should tri-state or disable the pad for maximum power savings. Only channel A map is shown. For multi-channel products the other channels are similar.

2.9.4.1 DDR Chip to Pad Mapping

The relationship of chip to pad drivers is shown in the following table. BIOS should disable or power down unused chips for maximum power savings.

Table 22: DDR Chip to Pad Mapping (DDR3 Mode)

Chiplet	Timing Group	Pad Number[1]	Pad[2]
ABYTE[0]	0	3, 2. 1, 0	MEMCKE0[2], MEMCKE0[3], MEMCKE0[0], MEMCKE0[1]
	1	Unused	
	2	11,10,9, 8, 7,6,5,4	MEMADD0[14,15,12], MEMBNK0[2], MEMADD0[8,7,11,9]
ABYTE[1]	0	3, 2, 1, 0	MEMCLK0_L[2], MEMCLK0_H[2], MEMCLK0_L[0], MEMCLK0_H[0]
	1	9,8,6,5,4 11, 10	MEMADD0[5,6,2,1,3,4] Unused, Unused
	2	Unused	
ABYTE[2]	0	3, 2, 1, 0	MEMCLK0_L[3], MEMCLK0_H[3], MEMCLK0_L[1], MEMCLK0_H[1]
	1	7, 6, 5, 4	MEMADD0[10], MEMBANK0[0], MEMADD0[0], MEMBANK0[1]
	2	11, 10, 9, 8	MEMCS0_L[6], MEMCS0_L[4], MEMCS0_L[2], MEMCS0_L[0]

Table 22: DDR Chip to Pad Mapping (DDR3 Mode) (Continued)

ABYTE[3]	0	3, 2, 1, 0	MEMADD0[13], MEMCAS0_L, MEMWE0_L, MEMRAS0_L
	1	7, 6, 5, 4	MEMODT0[3], MEMODT0[1], MEMODT0[0], MEMODT0[2]
	2	11, 10, 9, 8	MEMCS0_L[7], MEMCS0_L[5], MEMCS0_L[3], MEMCS0_L[1]
DBYTE[0][3]	0	3,2,1,0	MEMDATA0[3,2,1,0]
	1	7,6,5,4	MEMDATA0[7,6,5,4]
	2	10, 8	MEMDQS_L[0], MEMDQS_H[0]
	2 for TX 3 for RX	11, 9	MEMDQS_L[8], MEMDQSDM[0]
DBYTE[8][3]	0	3,2,1,0	MEMCHECK0[3,2,1,0]
	1	7,6,5,4	MEMCHECK0[7,6,5,4]
	2	10, 8	MEMCQS_L[0], MEMCQS_H[0]
	2 for TX 3 for RX	11, 9	MEMCQS_L[1], MEMCQSDM[0]

1. Pad number is the logical address of a CSR addressable timing or impedance control.
 Only channel A map is shown. For multi-channel products the other channels are similar.
2. Pad column shows pads only for DBYTE[0]. Data chips [7:1] are repeated with sequential DQ/DQS/DM pin numbers.
3. MEMDQSDM functions as DM when using unbuffered DIMMs. MEMDQS_H is the positive polarity of the “lower nibble” DQS pad used for all devices.

Table 23: DDR Chip to Pad Mapping (DDR4 Mode)

Chiplet	Timing Group	Pad Number[1]	Pad[2]

Table 23: DDR Chip to Pad Mapping (DDR4 Mode) (Continued)

ABYTE[0]	0	3, 2. 1, 0	MEMCKE0[2], MEMCKE0[3], MEMCKE0[0], MEMCKE0[1]
	1	Unused	
	2	11, 10, 9, 8, 7,6,5,4	MEMADD0[14]/BG1, MEMADD0[15]/ACT_L MEMADD0[12], MEMBNK0[2]/BG0, MEMADD0[8,7,11,9]
ABYTE[1]	0	3, 2, 1, 0	MEMCLK0_L[2], MEMCLK0_H[2], MEMCLK0_L[0], MEMCLK0_H[0]
	1	9,8,6,5,4 11,10	MEMADD0[5,6,2,1,3,4] Unused, Unused
	2	Unused	
ABYTE[2]	0	3, 2, 1, 0	MEMCLK0_L[3], MEMCLK0_H[3], MEMCLK0_L[1], MEMCLK0_H[1]
	1	7, 6, 5, 4	MEMADD0[10], MEMBANK0[0], MEMADD0[0], MEMBANK0[1]
	2	11, 10, 9, 8	MEMCS0_L[6], MEMCS0_L[4], MEMCS0_L[2], MEMCS0_L[0]
ABYTE[3]	0	3, 2, 1, 0	MEMADD0[13], MEMCAS0_L/ADD[15], MEMWE0_L/ADD[14], MEMRAS0_L/ADD[16]
	1	7, 6, 5, 4	MEMODT0[3], MEMODT0[1], MEMODT0[0], MEMODT0[2]
	2	11, 10, 9, 8	MEMCS0_L[7], MEMCS0_L[5], MEMCS0_L[3], MEMCS0_L[1]

Table 23: DDR Chip to Pad Mapping (DDR4 Mode) (Continued)

DBYTE[0][3]	0	3,2,1,0	MEMDATA0[3,2,1,0]
	1	7,6,5,4	MEMDATA0[7,6,5,4]
	2	10, 8	MEMDQS_L[0], MEMDQS_H[0]
	2 for TX 3 for RX	11, 9	MEMDQS_L[8], MEMDQSDM[0]
DBYTE[8][3]	0	3,2,1,0	MEMCHECK0[3,2,1,0]
	1	7,6,5,4	MEMCHECK0[7,6,5,4]
	2	10, 8	MEMCQS_L[0], MEMCQS_H[0]
	2 for TX 3 for RX	11, 9	MEMCQS_L[1], MEMCQSDM[0]

1. Pad number is the logical address of a CSR addressable timing or impedance control.
 Only channel A map is shown. For multi-channel products the other channels are similar.
2. Pad column shows pads only for DBYTE[0]. Data chips [7:1] are repeated with sequential DQ/DQS/DM pin numbers.
3. MEMDQSDM functions as DM when using unbuffered DIMMs. MEMDQS_H is the positive polarity of the “lower nibble” DQS pad used for all devices.

2.9.5 DRAM Controller Direct Response Mode

The DCT supports direct response mode for responding to a cache line fill request before the DCT is initialized. In direct response mode, the target DCT responds to a cache line fill request by returning 64 bytes of the repeated value 90h without issuing a read transaction on the DRAM bus. The BIOS uses this feature to allocate cache lines for temporary data storage. The controller exits direct response mode when either D18F2x78_dct[0][ChanVal] is set to 1. See 2.3.3 [Using L2 Cache as General Storage During Boot].

2.9.6 DRAM Data Burst Mapping

DRAM requests are mapped to data bursts on the DDR bus in the following order:
- A 64 byte request is mapped to each of the eight sequential data beats as QW0, QW1, ... QW7.

2.9.7 SOC Specific Definitions

Table 24: DCT Definitions

Term	Definition
SRAMMsgBlk	SRAM message block. See 2.9.8.3 [SRAM Message Block].

2.9.8 PMU

The processor includes a phy micro-controller unit (PMU) used for training the DDR data bus during boot. BIOS communicates with the PMU using one of two methods.

- Mailbox: BIOS waits for upstream messages from the PMU for action synchronization or for status messages.
 - A 16-bit mailbox exists for upstream messages. See 2.9.8.1.
 - A 16-bit mailbox exists for upstream data transfer. See 2.9.8.2.
- SRAM: BIOS may read or write the memory used by the PMU to send or receive complex message data. The PMU must be halted or in the reset state for BIOS to access PMU SRAM.

2.9.8.1 Upstream Mailbox1 Message Pending

To wait for an upstream message BIOS does the following:

1. Wait until D18F2x9C_x0002_0004_dct[0][UsRdy]=0.
2. Read D18F2x9C_x0002_0032_dct[0][Message]. See Table 25 for a list of message names and values.
3. Program D18F2x9C_x0002_0033_dct[0][Rdy] = 1.

See 2.9.9.5 for related information.

Table 25: Upstream Mailbox 1 Messages

Name	D18F2x9C_x0002_0032_dct[0] [Message]	Description
DevInit	00h	PMU has completed DevInit.
TrainWrLvl	01h	PMU has completed TSTAGE_WrLvl.
TrainRxEn	02h	PMU has completed TSTAGE_RxEn.
TrainRdDqs1D	03h	PMU has completed TSTAGE_RdDqs1D.
TrainWrDq1D	04h	PMU has completed TSTAGE_WrDq1D.
TrainRd2D	05h	PMU has completed TSTAGE_Rd2D.
TrainWr2D	06h	PMU has completed TSTAGE_Wr2D.
PMUQEmpty	07h	PMU has completed all of its SequenceCtl tasks and is in a powergated idle state.
US2MsgRdy	08h	PMU is ready to stream a message through US mailbox 2.
FAIL	0FFh	PMU has encountered an error which requires requester to abort waiting for remaining pending upstream messages.

2.9.8.2 Upstream Mailbox2 Message Pending

To receive a block of data through US mailbox 2, BIOS does the following:

1. Wait until D18F2x9C_x0002_0004_dct[0][Us2Rdy] = 0.
2. Read D18F2x9C_x0002_0034_dct[0][Message]. The first item received is the COUNT.
3. Program D18F2x9C_x0002_0035_dct[0][Rdy] = 1.
4. Decrement COUNT and loop to step 1 until COUNT = -1.

2.9.8.3 SRAM Message Block

The SRAM message block is used to pass information from BIOS to PMU and vice-versa. BIOS accesses the SRAM message block through D18F2x9C_x0005_[0FFF:0000]_dct[0].

2.9.9 DCT/DRAM Initialization and Resume

DRAM initialization involves several steps in order to configure the DRAM controllers and the DRAM, and to tune the DRAM channel for optimal performance. DRAM resume requires several steps to configure the DCTs to properly resume from the S3 state. The following sequence describes the steps needed after a reset for initialization or resume:

- To disable an unused DRAM channel see 2.9.9.8.

1. Configure the DDR supply voltage regulator. See 2.9.9.1.
2. NB P-state specific initialization.
 A. Program the DCT configuration registers D18F2x1F[C:0]_dct[0] for each NB P-state.
 B. Force NB P-state to NBP0. See 2.5.3.1.2.2.
3. DDR phy initialization. See 2.9.9.2.
4. If BIOS is booting from an unpowered state (ACPI S4, S5 or G3), then it performs the following:
 A. Program DRAM controller general configuration, for both memory P-states. See 2.9.9.3.
 B. Program DCT specific configuration for training, for both memory P-states. See 2.9.9.4.
 C. Program the remaining DCT registers not covered by an explicit sequence dependency.
 D. Program D18F2x9C_x0002_0060_dct[0][MemReset_L] = 0.
 E. Program D18F2x9C_x0002_000B_dct[0] = 0004h.
 F. Program D18F2x9C_x0002_000B_dct[0] = 0000h.
 G. Perform DRAM device initialization and data training. See 2.9.9.5.
 H. Enable phy auto-calibration. See 2.9.9.2.8.3.
 I. Synchronous channel initialization. See 2.9.9.7.
 J. Program D18F2x90_dct[0][ExitSelfRef] = 1. Wait for D18F2x90_dct[0][ExitSelfRef] = 0. Wait Tref*2.
 K. Program D18F2x78_dct[0][ChanVal] = 1.
 L. NB P-state specific training. For each NB P-state from NBP0 to D18F5x170[NbPstateMaxVal]:
 a. Force the NB P-state. See 2.5.3.1.2.2.
 b. MaxRdLatency training. See 2.9.9.6.1.
 M. Program DCT specific configuration for normal operation, for both memory P-states. See 2.9.9.4.
 N. Program DRAM phy for power savings. See 2.9.9.9.

Else If BIOS is resuming the platform from S3 state, then it performs the following:

 A. Restore all DCT and phy registers that were programmed during the first boot from non-volatile storage. See 2.9.9.3, 2.9.9.4, and 2.9.9.9 for a review of registers.

B. Restore the trained delayed values from nonvolatile storage. See 2.9.9.2.10.
C. Program D18F2x9C_x0002_0060_dct[0][MemReset_L] = 1.
D. Program D18F2x9C_x0002_000B_dct[0] = 0004h.
E. Program D18F2x9C_x0002_000B_dct[0] = 0000h.
F. Fence the CalOnce. See 2.9.9.2.8.2.
G. Enable phy auto-calibration. See 2.9.9.2.8.3.
H. Synchronous channel initialization. See 2.9.9.7.
I. Program D18F2x90_dct[0][ExitSelfRef] = 1. Wait for D18F2x90_dct[0][ExitSelfRef] == 0.
J. Program D18F2x78_dct[0][ChanVal] = 1.

5. Release NB P-state force. See 2.5.3.1.2.2.

The DRAM subsystem is ready for use.

2.9.9.1 Low Voltage

For DDR3 devices, the processor supports JEDEC defined 1.5V, 1.35V and 1.25V devices.

For DDR4 devices, the processor supports JEDEC defined 1.2V devices.

Platforms supporting DDR3 low voltage devices should power up VDDIO at 1.35V. BIOS should not operate DIMMs at voltages higher than supported. For DDR3 this is indicated by SPD Byte 6: Module Nominal Voltage, VDD.

BIOS should consult vendor data sheets for the supply voltage regulator programming requirements. On supported platforms, BIOS must take steps to configure the supply voltage regulators as follows:

1. Read the SPD of all devices within the programmable VDDIO domain and check all of the defined bits within the SPD byte to determine the common operating voltages.
2. Configure VDDIO to match the lowest common supported voltage based on the SPD values.
 - If the DIMMs do not specify a common operating voltage then BIOS must take platform vendor defined action to notify the end user of the mismatch and to protect DIMMs from damage.
3. Additional derating of the DDR speed may be necessary for reliable operation at lower voltage.

2.9.9.2 DDR Phy Initialization

The BIOS initializes the phy and the internal interface from the DCT to the phy after each reset and for each time a MEMCLK frequency change is made.

BIOS obtains size, loading, and frequency information about the DIMMs and channels using SPDs prior to phy initialization. BIOS then performs the following actions:

1. Program D18F2x9C_x0002_0099_dct[0][PmuReset,PmuStall] = {1,1}.
2. Program D18F2x9C_x0002_000E_dct[0][PhyDisable] = 0.
3. According to the type of DRAM attached, program the following phy mode:
 - D18F2x9C_x0000_000E_dct[0][D4_Mode]
 - D18F2x9C_x00[F,8:0]1_000E_dct[0][D4_Mode]
 - D18F2x9C_x0002_000E_dct[0][D4_Mode]
 - D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0][MajorMode]
 - D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]043_dct[0][MajorMode] = Ddr4Mode ? 011b : (M1 State ? 100b : 000b)
4. Program general phy static configuration. See 2.9.9.2.1.

5. Phy Voltage Level Programming. See 2.9.9.2.2.
6. Program auto-calibration. See 2.9.9.2.8.
7. Program DRAM channel frequency. See 2.9.9.2.3.
8. Program default CAD bus values. See 2.9.9.2.4.
9. Program default data bus values. See 2.9.9.2.5.
10. Program FIFO pointer init values. See 2.9.9.2.6.
11. Program predriver values. See 2.9.9.2.7.
12. Program D18F2x9C_x0002_0033_dct[0][Rdy] = 1.
13. Program D18F2x9C_x0002_0035_dct[0][Rdy] = 1.

2.9.9.2.1 Phy General Configuration

BIOS programs the following according to the static configuration:

- If Ddr3Mode or Ddr4Mode:
 - Program D18F2x9C_x00[F,1:0]4_00E[7:0]_dct[0][OdtRdPatCs, OdtWrPatCs] per 2.9.9.3.1 [DRAM ODT Pin Control].
 - Program D18F2x9C_x00[F,1:0]4_00E[7:0]_dct[0] per D18F2x[23C:238]_dct[0] and D18F2x[234:230]_dct[0].
 - Program D18F2x9C_x00[F,8:0]1_0016_dct[0][POdtWidthDq, POdtWidthDqs, POdtStartDelayDq, POdtStartDelayDqs, ProcOdtOff, ProcOdtOn].
 - Program D18F2x9C_x00F0_0015_dct[0][VrefFilt] = 0000b.
 - Program D18F2x9C_x0002_0098_dct[0][CalD4D3] = 0.

BIOS programs the following for maximum power savings prior to training:

- Program D18F2x9C_x03F6_F04E_dct[0] = 0000h.
 - See D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0] and D18F2x9C_x00[F,8:0]1_[F,B:0]04E_dct[0].
- Program D18F2x9C_x03F6_F050_dct[0] = 8000h.
 - See D18F2x9C_x00[F,3:0]0_[F,B:0]050_dct[0] and D18F2x9C_x00[F,8:0]1_[F,B:0]050_dct[0].
- Program D18F2x9C_x00[F,3:0]0_001A_dct[0] = 0022h.
- Program D18F2x9C_x00[F,8:0]1_001A_dct[0] = 0033h.
- Program D18F2x9C_x0002_001A_dct[0] = 0001h.
- Program D18F2x9C_x0002_005A_dct[0] = 0000h if FP4 package.
- Program D18F2x9C_x0002_005B_dct[0] = 0000h if no EVENT pin thermal management.
 - See 2.9.14 [DRAM On DIMM Thermal Management and Power Capping].
- Program D18F2x9C_x00F1_F051_dct[0] = 0052h.
 - See D18F2x9C_x00[F,8:0]1_[F,B:0]051_dct[0].
- If D18F2x9C_x0[3,1:0][F,8:0]1_0010_dct[0][X4Dimm] == 0 then program:
 - D18F2x9C_x00F1_0077_dct[0][DllAltPiClkOutEn] = 1.
 - D18F2x9C_x00F1_0277_dct[0][DllAltPiClkInEn] = 1.
 - D18F2x9C_x00F1_0277_dct[0][DllPowerDown] =1.
 - See D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0].

2.9.9.2.2 Phy Voltage Level Programming

BIOS programs the following according to the desired phy VDDIO voltage level:

- Program D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0][AVoltageLevel].
- Program D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0][VoltageLevel].

- Program D18F2x9C_x0002_009B_dct[0] = E024h.
- Program D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]044_dct[0] = (Ddr4Mode & DimmsPopulated > 1) ? 5Fh : 40h).
- Program D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]045_dct[0] = (Ddr4Mode & DimmsPopulated > 1) ? 5Fh : 40h).

See 2.9.9.1 [Low Voltage].

2.9.9.2.3 DRAM Channel Frequency

BIOS programs the DCT and the phy according to the data rate. BIOS must ensure that the DCT and the phy operate at a matched rate prior to normal operations. See D18F2x94_dct[0][MemClkFreq], D18F2x2E0_dct[0][M1MemClkFreq].

To program a new rate in the phy:

1. Program D18F2x9C_x0002_0093_dct[0][PllRegWaitTime] = 4Bh.
2. Program D18F2x9C_x0002_0089_dct[0][PllLockTime] = C8h.
3. Program D18F2x9C_x0002_0000_dct[0][PllMultDiv].
4. Program D18F2x9C_x0002_0080_dct[3:0][PMUClkDiv].
 - See D18F2x9C_x0[3,1:0]02_0080_dct[0].
5. Program D18F2x9C_x0002_0001_dct[0][PllMultDiv] = If Ddr4Mode Then ‘1333 MT/s’ else ‘667 MT/s’.
6. Program D18F2x9C_x0102_0080_dct[3:0][PMUClkDiv].
 - See D18F2x9C_x0[3,1:0]02_0080_dct[0].

The new settings takes effect after BIOS or the DCT requests a memory P-state change request via D18F2x9C_x0002_000B_dct[0]. See 2.9.9 [DCT/DRAM Initialization and Resume].

2.9.9.2.4 DRAM Address, Command, and Output Driver Control

This section describes the settings for programming the timing on the DDR address and command pins. The following tables document the address timing and output driver settings on a per channel basis. These tables document the optimal settings for motherboards which meet the relevant motherboard design guidelines.

- Only the value for a single control unit register is described. The values in the table should be broadcast to all instances of registers of the same control unit type, unless otherwise noted.

Table 26: BIOS Recommendations for DDR3 SODIMM Address and Command Bus

Condition					D18F2x94_dct[0] [SlowAccessMode]	{00h,00b,AddrCmd-Setup[5:0],00b,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Num-Dimm Slots	DdrRate	VDDIO	DIMM 0	DIMM 1						
1	667	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	667	1.25, 1.35, 1.5	DR	-	0	003B0000h	1Fh	1Fh	1Fh	1Fh
1	800	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	800	1.25, 1.35, 1.5	DR	-	0	003B0000h	1Fh	1Fh	1Fh	1Fh
1	1066	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1066	1.25, 1.35, 1.5	DR	-	0	00380000h	1Fh	1Fh	1Fh	1Fh
1	1333	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1333	1.25, 1.35, 1.5	DR	-	0	00360000h	1Fh	1Fh	1Fh	1Fh
1	1600	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1600	1.25, 1.35, 1.5	DR	-	1	00000000h	1Fh	1Fh	1Fh	1Fh
1	1866	1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1866	1.35, 1.5	DR	-	1	00000000h	1Fh	1Fh	1Fh	1Fh
1	2133	1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	2133	1.5	DR	-	1	00000000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	NP	DR	0	003B0000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	DR	NP	0	003B0000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	SR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	667	1.25, 1.35, 1.5	DR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	667	1.25, 1.35, 1.5	DR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	667	1.25, 1.35, 1.5	SR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	800	1.25, 1.35, 1.5	NP	DR	0	003B0000h	1Fh	1Fh	1Fh	1Fh
2	800	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	800	1.25, 1.35, 1.5	DR	NP	0	003B0000h	1Fh	1Fh	1Fh	1Fh

Table 26: BIOS Recommendations for DDR3 SODIMM Address and Command Bus (Continued)

Condition					D18F2x94_dct[0] [SlowAccessMode]	{00h,00b,AddrCmd-Setup[5:0],00b,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Num-Dimm Slots	DdrRate	VDDIO	DIMM 0	DIMM 1						
2	800	1.25, 1.35, 1.5	SR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	DR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	DR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	SR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	NP	DR	0	00380000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	DR	NP	0	00380000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	SR	SR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	DR	DR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	DR	SR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	SR	DR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	NP	DR	0	00360000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	DR	NP	0	00360000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	SR	SR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	DR	DR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	DR	SR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	SR	DR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1600	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.25, 1.35, 1.5	NP	DR	1	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.25, 1.35, 1.5	DR	NP	1	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.35, 1.5	SR	SR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh
2	1600	1.35, 1.5	DR	DR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh

Table 26: BIOS Recommendations for DDR3 SODIMM Address and Command Bus (Continued)

Condition					D18F2x94_dct[0] [SlowAccessMode]	{00h,00b,AddrCmd-Setup[5:0],00b,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Num-Dimm Slots	DdrRate	VDDIO	DIMM 0	DIMM 1						
2	1600	1.35, 1.5	DR	SR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh
2	1600	1.35, 1.5	SR	DR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh
2	1866	1.35, 1.5	NP	SR	0	003C3C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.35, 1.5	NP	DR	1	00003C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.35, 1.5	SR	NP	0	003C3C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.35, 1.5	DR	NP	1	00003C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.5	SR	SR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	1866	1.5	DR	DR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	1866	1.5	DR	SR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	1866	1.5	SR	DR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	2133	1.5	NP	SR	0	003B3B3Bh	1Fh	1Fh	1Fh	1Fh
2	2133	1.5	NP	DR	1	00003B3Bh	1Fh	1Fh	1Fh	1Fh
2	2133	1.5	SR	NP	0	003B3B3Bh	1Fh	1Fh	1Fh	1Fh
2	2133	1.5	DR	NP	1	00003B3Bh	1Fh	1Fh	1Fh	1Fh

1. BIOS writes the values for AddrCmdSetup, CsOdtSetup, and CkeSetup into the SRAMMsgBlk for each memory P-state.
2. BIOS programs DrvStrenP = DrvStrenN for each instance.

Table 27: BIOS Recommendations for DDR3 UDIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0] [SlowAccessMode]	{0000000000b ,AddrCmd-Setup[5:0],00b ,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Condi-tion:NumDimmS-lots	Condi-tion:DdrRate	Condi-tion:VDDIO	Condi-tion:DIMM0	Condi-tion:DIMM1						
1	667	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	667	1.25, 1.35, 1.5	DR	-	0	003B0000h	1Fh	1Fh	1Fh	1Fh
1	800	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	800	1.25, 1.35, 1.5	DR	-	0	003B0000h	1Fh	1Fh	1Fh	1Fh
1	1066	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1066	1.25, 1.35, 1.5	DR	-	0	00380000h	1Fh	1Fh	1Fh	1Fh
1	1333	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1333	1.25, 1.35, 1.5	DR	-	0	00360000h	1Fh	1Fh	1Fh	1Fh
1	1600	1.25, 1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1600	1.25, 1.35, 1.5	DR	-	1	00000000h	1Fh	1Fh	1Fh	1Fh
1	1866	1.35, 1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	1866	1.35, 1.5	DR	-	1	00000000h	1Fh	1Fh	1Fh	1Fh
1	2133	1.5	SR	-	0	00000000h	1Fh	1Fh	1Fh	1Fh
1	2133	1.5	DR	-	1	00000000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	NP	DR	0	003B0000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	DR	NP	0	003B0000h	1Fh	1Fh	1Fh	1Fh
2	667	1.25, 1.35, 1.5	SR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	667	1.25, 1.35, 1.5	DR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	667	1.25, 1.35, 1.5	DR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	667	1.25, 1.35, 1.5	SR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	800	1.25, 1.35, 1.5	NP	DR	0	003B0000h	1Fh	1Fh	1Fh	1Fh
2	800	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	800	1.25, 1.35, 1.5	DR	NP	0	003B0000h	1Fh	1Fh	1Fh	1Fh

Table 27: BIOS Recommendations for DDR3 UDIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0] [SlowAccessMode]	{0000000000b,AddrCmd-Setup[5:0],00b,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1						
2	800	1.25, 1.35, 1.5	SR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	DR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	DR	SR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	800	1.25, 1.35, 1.5	SR	DR	0	00390039h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	NP	DR	0	00380000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	DR	NP	0	00380000h	1Fh	1Fh	1Fh	1Fh
2	1066	1.25, 1.35, 1.5	SR	SR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	DR	DR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	DR	SR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1066	1.25, 1.35, 1.5	SR	DR	0	00350037h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	NP	DR	0	00360000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	DR	NP	0	00360000h	1Fh	1Fh	1Fh	1Fh
2	1333	1.25, 1.35, 1.5	SR	SR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	DR	DR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	DR	SR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1333	1.25, 1.35, 1.5	SR	DR	1	00000035h	1Fh	1Fh	3Fh	1Fh
2	1600	1.25, 1.35, 1.5	NP	SR	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.25, 1.35, 1.5	NP	DR	1	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.25, 1.35, 1.5	SR	NP	0	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.25, 1.35, 1.5	DR	NP	1	00000000h	1Fh	1Fh	1Fh	1Fh
2	1600	1.35, 1.5	SR	SR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh
2	1600	1.35, 1.5	DR	DR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh

Table 27: BIOS Recommendations for DDR3 UDIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0] [SlowAccessMode]	{0000000000b,AddrCmdSetup[5:0],00b,CsOdtSetup[5:0], 00b, CkeSetup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1						
2	1600	1.35, 1.5	DR	SR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh
2	1600	1.35, 1.5	SR	DR	1	0000002Bh	1Fh	1Fh	3Fh	1Fh
2	1866	1.35, 1.5	NP	SR	0	003C3C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.35, 1.5	NP	DR	1	00003C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.35, 1.5	SR	NP	0	003C3C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.35, 1.5	DR	NP	1	00003C3Ch	1Fh	1Fh	1Fh	1Fh
2	1866	1.5	SR	SR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	1866	1.5	DR	DR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	1866	1.5	DR	SR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	1866	1.5	SR	DR	1	00000031h	1Fh	1Fh	3Fh	1Fh
2	2133	1.5	NP	SR	0	003B3B3Bh	1Fh	1Fh	1Fh	1Fh
2	2133	1.5	NP	DR	1	00003B3Bh	1Fh	1Fh	1Fh	1Fh
2	2133	1.5	SR	NP	0	003B3B3Bh	1Fh	1Fh	1Fh	1Fh
2	2133	1.5	DR	NP	1	00003B3Bh	1Fh	1Fh	1Fh	1Fh

1. BIOS writes the values for AddrCmdSetup, CsOdtSetup, and CkeSetup into the SRAMMsgBlk for each memory P-state.
2. BIOS programs DrvStrenP = DrvStrenN for each instance.

Table 28: BIOS Recommendations for DDR4 UDIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0] [SlowAccessMode]	{0000000000b ,AddrCmd-Setup[5:0],00b ,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Condi-tion:NumDimmS-lots	Condi-tion:DdrRate	Condi-tion:VDDIO	Condi-tion:DIMM0	Condi-tion:DIMM1						
1	1333 + 1600 + 1866 + 2133 + 2400	1.2	SR	NP	0	0x00000000	0x1F	0x1F	0x1F	0x1F
1	1333	1.2	DR	NP	0	0x00360000	0x1F	0x1F	0x1F	0x1F
1	1600 + 1866 + 2133 + 2400	1.2	DR	NP	1	0x00000000	0x1F	0x1F	0x1F	0x1F
2	1333 + 1600 + 1866 + 2133 + 2400	1.2	NP	SR	0	0x00000000	0x1F	0x1F	0x1F	0x1F
2	1333	1.2	NP	DR	0	0x00360000	0x1F	0x1F	0x1F	0x1F
2	1333 + 1600 + 1866 + 2133 + 2400	1.2	SR	NP	0	0x00000000	0x1F	0x1F	0x1F	0x1F
2	1333 + 1600 + 1866	1.2	SR + DR	SR + DR	1	0x00000000	0x1F	0x1F	0x3F	0x1F
2	1333	1.2	DR	NP	0	0x00360000	0x1F	0x1F	0x1F	0x1F
2	1600 + 1866 + 2133	1.2	NP	DR	1	0x00000000	0x1F	0x1F	0x1F	0x1F

Table 28: BIOS Recommendations for DDR4 UDIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0] [SlowAccessMode]	{0000000000b,AddrCmdSetup[5:0],00b,CsOdtSetup[5:0], 00b, CkeSetup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1						
2	1600 + 1866 + 2133	1.2	DR	NP	1	0x00000000	0x1F	0x1F	0x1F	0x1F
2	2133	1.2	SR	SR	1	0x00000000	0x1F	0x1F	0x3F	0x1F

1. BIOS writes the values for AddrCmdSetup, CsOdtSetup, and CkeSetup into the SRAMMsgBlk for each memory P-state.
2. BIOS programs DrvStrenP = DrvStrenN for each instance.

Table 29: BIOS Recommendations for DDR4 SODIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0] [SlowAccessMode]	{0000000000b ,AddrCmd-Setup[5:0],00b ,CsOdt-Setup[5:0], 00b, Cke-Setup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] [DrvStrenN][2] for CLK
Condi-tion:NumDimmS-lots	Condi-tion:DdrRate	Condi-tion:VDDIO	Condi-tion:DIMM0	Condi-tion:DIMM1						
1	1333 + 1600 + 1866 + 2133 + 2400	V1_2	SR	NP	0	0x00000000	0x1F	0x1F	0x1F	0x1F
1	1333	V1_2	DR	NP	0	0x00360000	0x1F	0x1F	0x1F	0x1F
1	1600 + 1866 + 2133 + 2400	V1_2	DR	NP	1	0x00000000	0x1F	0x1F	0x1F	0x1F
2	1333 + 1600 + 1866 + 2133	V1_2	NP	SR	0	0x00000000	0x1F	0x1F	0x1F	0x1F
2	1333	V1_2	NP	DR	0	0x00360000	0x1F	0x1F	0x1F	0x1F
2	1333 + 1600 + 1866 + 2133	V1_2	SR	NP	0	0x00000000	0x1F	0x1F	0x1F	0x1F
2	1333 + 1600 + 1866	V1_2	SR + DR	SR + DR	1	0x00000000	0x1F	0x1F	0x3F	0x1F
2	1333	V1_2	DR	NP	0	0x00360000	0x1F	0x1F	0x1F	0x1F
2	1600 + 1866 + 2133	V1_2	NP	DR	1	0x00000000	0x1F	0x1F	0x1F	0x1F

Table 29: BIOS Recommendations for DDR4 SODIMM Address and Command Bus Configuration

Condition					D18F2x94_dct[0][SlowAccessMode]	{0000000000b,AddrCmdSetup[5:0],00b,CsOdtSetup[5:0],00b,CkeSetup[5:0]}[1]	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0][DrvStrenN][2] for CKE	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0][DrvStrenN][2] for CS and ODT	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0][DrvStrenN][2] for AddrCmd	D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0][DrvStrenN][2] for CLK
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1						
2	1600 + 1866 + 2133	V1_2	DR	NP	1	0x00000000	0x1F	0x1F	0x1F	0x1F
2	2133	V1_2	SR	SR	1	0x00000000	0x1F	0x1F	0x3F	0x1F

1. BIOS writes the values for AddrCmdSetup, CsOdtSetup, and CkeSetup into the SRAMMsgBlk for each memory P-state.
2. BIOS programs DrvStrenP = DrvStrenN for each instance.

2.9.9.2.5 DRAM Data Bus Configuration

This section describes the settings required for programming the drive settings and slew rates on the data bus pins. The following tables document the data bus values on a per channel basis. DIMM0 is the DIMM closest to the processor on that channel and DIMM1 is the DIMM farthest from the processor on that channel. DIMMs must be populated from farthest slot to closest slot to the processor on a per channel basis (when a daisy chain topology is used). Populations that are not shown in these tables are not supported. These tables document the optimal settings for motherboards which meet the relevant motherboard design guidelines.

- Program D18F2x9C_x0002_0087_dct[0][DisAutoComp, DisPredriverCal] = {1,1}.
- Only the value for a single control unit register is described. The values in the table should be broadcast to all instances of registers of the same control unit type, unless otherwise noted.
- Program D18F2x9C_x0002_0087_dct[0][DisAutoComp] = 0.

Table 30: BIOS Recommendations for DDR3 SODIMM Data Bus Configuration

Condition					RTT_Nom	RTT_Wr	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1					
1	667	1.25, 1.35, 1.5	SR	-	120	Off	70h	70h	01h
1	667	1.25, 1.35, 1.5	DR	-	120	Off	70h	70h	01h
1	800	1.25, 1.35, 1.5	SR	-	120	Off	70h	70h	01h
1	800	1.25, 1.35, 1.5	DR	-	120	Off	70h	70h	01h
1	1066	1.25, 1.35, 1.5	SR	-	120	Off	70h	70h	04h
1	1066	1.25, 1.35, 1.5	DR	-	120	Off	70h	70h	04h
1	1333	1.25, 1.35, 1.5	SR	-	60	Off	70h	70h	05h
1	1333	1.25, 1.35, 1.5	DR	-	60	Off	70h	70h	05h
1	1600	1.25, 1.35, 1.5	SR	-	60	Off	70h	70h	0Ch
1	1600	1.25, 1.35, 1.5	DR	-	40	Off	70h	70h	0Ch
1	1866	1.35, 1.5	SR	-	40	Off	70h	70h	0Ch
1	1866	1.35, 1.5	DR	-	40	Off	70h	70h	0Ch
1	2133	1.5	SR	-	40	Off	70h	70h	0Ch
1	2133	1.5	DR	-	40	Off	70h	70h	0Ch
2	667	1.25, 1.35, 1.5	NP	SR	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	NP	DR	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	SR	NP	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	DR	NP	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	SR	SR	40	120	75h	75h	04h
2	667	1.25, 1.35, 1.5	DR	DR	40	120	75h	75h	04h
2	667	1.25, 1.35, 1.5	DR	SR	40	120	75h	75h	04h
2	667	1.25, 1.35, 1.5	SR	DR	40	120	75h	75h	04h
2	800	1.25, 1.35, 1.5	NP	SR	120	Off	70h	70h	01h
2	800	1.25, 1.35, 1.5	NP	DR	120	Off	70h	70h	01h
2	800	1.25, 1.35, 1.5	SR	NP	120	Off	70h	70h	01h
2	800	1.25, 1.35, 1.5	DR	NP	120	Off	70h	70h	01h

Table 30: BIOS Recommendations for DDR3 SODIMM Data Bus Configuration (Continued)

Condition					RTT_Nom	RTT_Wr	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1					
2	800	1.25, 1.35, 1.5	SR	SR	40	120	75h	75h	05h
2	800	1.25, 1.35, 1.5	DR	DR	40	120	75h	75h	05h
2	800	1.25, 1.35, 1.5	DR	SR	40	120	75h	75h	05h
2	800	1.25, 1.35, 1.5	SR	DR	40	120	75h	75h	05h
2	1066	1.25, 1.35, 1.5	NP	SR	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	NP	DR	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	SR	NP	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	DR	NP	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	SR	SR	40	120	75h	75h	0Ch
2	1066	1.25, 1.35, 1.5	DR	DR	40	120	75h	75h	0Ch
2	1066	1.25, 1.35, 1.5	DR	SR	40	120	75h	75h	0Ch
2	1066	1.25, 1.35, 1.5	SR	DR	40	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	NP	SR	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	NP	DR	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	SR	NP	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	DR	NP	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	SR	SR	30	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	DR	DR	30	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	DR	SR	30	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	SR	DR	30	120	75h	75h	0Ch
2	1600	1.25, 1.35, 1.5	NP	SR	40	Off	70h	70h	0Ch
2	1600	1.25, 1.35, 1.5	NP	DR	40	Off	70h	70h	0Ch
2	1600	1.25, 1.35, 1.5	SR	NP	40	Off	70h	70h	0Ch
2	1600	1.25, 1.35, 1.5	DR	NP	40	Off	70h	70h	0Ch
2	1600	1.35, 1.5	SR	SR	20	60	75h	75h	0Ch
2	1600	1.35, 1.5	DR	DR	20	60	75h	75h	0Ch

Table 30: BIOS Recommendations for DDR3 SODIMM Data Bus Configuration (Continued)

Condition					RTT_Nom	RTT_Wr	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1					
2	1600	1.35, 1.5	DR	SR	20	60	75h	75h	0Ch
2	1600	1.35, 1.5	SR	DR	20	60	75h	75h	0Ch
2	1866	1.35, 1.5	NP	SR	40	Off	70h	70h	0Ch
2	1866	1.35, 1.5	NP	DR	40	Off	70h	70h	0Ch
2	1866	1.35, 1.5	SR	NP	40	Off	70h	70h	0Ch
2	1866	1.35, 1.5	DR	NP	40	Off	70h	70h	0Ch
2	1866	1.5	SR	SR	20	60	75h	75h	0Ch
2	1866	1.5	DR	DR	20	60	75h	75h	0Ch
2	1866	1.5	DR	SR	20	60	75h	75h	0Ch
2	1866	1.5	SR	DR	20	60	75h	75h	0Ch
2	2133	1.5	NP	SR	40	Off	70h	70h	0Ch
2	2133	1.5	NP	DR	40	Off	70h	70h	0Ch
2	2133	1.5	SR	NP	40	Off	70h	70h	0Ch
2	2133	1.5	DR	NP	40	Off	70h	70h	0Ch

1. BIOS programs DrvStrenN = DrvStrenP for each instance.
 BIOS programs instances for “MEMDQSDM” (when used as a data mask) with the same value as DQ.
2. BIOS programs ODTStrenN = ODTStrenP for each instance.
 BIOS programs all used instances with these values.

Table 31: BIOS Recommendations for DDR3 UDIMM Data Bus Configuration

Condition					RTT_Nom	RTT_Wr	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1					
1	667	1.25, 1.35, 1.5	SR	-	120	Off	70h	70h	01h
1	667	1.25, 1.35, 1.5	DR	-	120	Off	70h	70h	01h
1	800	1.25, 1.35, 1.5	SR	-	120	Off	70h	70h	01h
1	800	1.25, 1.35, 1.5	DR	-	120	Off	70h	70h	01h
1	1066	1.25, 1.35, 1.5	SR	-	120	Off	70h	70h	04h
1	1066	1.25, 1.35, 1.5	DR	-	120	Off	70h	70h	04h
1	1333	1.25, 1.35, 1.5	SR	-	60	Off	70h	70h	05h
1	1333	1.25, 1.35, 1.5	DR	-	60	Off	70h	70h	05h
1	1600	1.25, 1.35, 1.5	SR	-	60	Off	70h	70h	0Ch
1	1600	1.25, 1.35, 1.5	DR	-	40	Off	70h	70h	0Ch
1	1866	1.35, 1.5	SR	-	40	Off	70h	70h	0Ch
1	1866	1.35, 1.5	DR	-	40	Off	70h	70h	0Ch
1	2133	1.5	SR	-	40	Off	70h	70h	0Ch
1	2133	1.5	DR	-	40	Off	70h	70h	0Ch
2	667	1.25, 1.35, 1.5	NP	SR	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	NP	DR	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	SR	NP	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	DR	NP	120	Off	70h	70h	01h
2	667	1.25, 1.35, 1.5	SR	SR	40	120	75h	75h	04h
2	667	1.25, 1.35, 1.5	DR	DR	40	120	75h	75h	04h
2	667	1.25, 1.35, 1.5	DR	SR	40	120	75h	75h	04h
2	667	1.25, 1.35, 1.5	SR	DR	40	120	75h	75h	04h
2	800	1.25, 1.35, 1.5	NP	SR	120	Off	70h	70h	01h
2	800	1.25, 1.35, 1.5	NP	DR	120	Off	70h	70h	01h
2	800	1.25, 1.35, 1.5	SR	NP	120	Off	70h	70h	01h
2	800	1.25, 1.35, 1.5	DR	NP	120	Off	70h	70h	01h

Table 31: BIOS Recommendations for DDR3 UDIMM Data Bus Configuration (Continued)

Condition					RTT_Nom	RTT_Wr	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1					
2	800	1.25, 1.35, 1.5	SR	SR	40	120	75h	75h	05h
2	800	1.25, 1.35, 1.5	DR	DR	40	120	75h	75h	05h
2	800	1.25, 1.35, 1.5	DR	SR	40	120	75h	75h	05h
2	800	1.25, 1.35, 1.5	SR	DR	40	120	75h	75h	05h
2	1066	1.25, 1.35, 1.5	NP	SR	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	NP	DR	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	SR	NP	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	DR	NP	120	Off	70h	70h	04h
2	1066	1.25, 1.35, 1.5	SR	SR	40	120	75h	75h	0Ch
2	1066	1.25, 1.35, 1.5	DR	DR	40	120	75h	75h	0Ch
2	1066	1.25, 1.35, 1.5	DR	SR	40	120	75h	75h	0Ch
2	1066	1.25, 1.35, 1.5	SR	DR	40	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	NP	SR	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	NP	DR	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	SR	NP	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	DR	NP	60	Off	70h	70h	05h
2	1333	1.25, 1.35, 1.5	SR	SR	30	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	DR	DR	30	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	DR	SR	30	120	75h	75h	0Ch
2	1333	1.25, 1.35, 1.5	SR	DR	30	120	75h	75h	0Ch
2	1600	1.25, 1.35, 1.5	NP	SR	40	Off	70h	70h	0Ch
2	1600	1.25, 1.35, 1.5	NP	DR	40	Off	70h	70h	0Ch
2	1600	1.25, 1.35, 1.5	SR	NP	40	Off	70h	70h	0Ch
2	1600	1.25, 1.35, 1.5	DR	NP	40	Off	70h	70h	0Ch
2	1600	1.35, 1.5	SR	SR	20	60	75h	75h	0Ch
2	1600	1.35, 1.5	DR	DR	20	60	75h	75h	0Ch

Table 31: BIOS Recommendations for DDR3 UDIMM Data Bus Configuration (Continued)

Condition					RTT_Nom	RTT_Wr	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
Condition:NumDimmSlots	Condition:DdrRate	Condition:VDDIO	Condition:DIMM0	Condition:DIMM1					
2	1600	1.35, 1.5	DR	SR	20	60	75h	75h	0Ch
2	1600	1.35, 1.5	SR	DR	20	60	75h	75h	0Ch
2	1866	1.35, 1.5	NP	SR	40	Off	70h	70h	0Ch
2	1866	1.35, 1.5	NP	DR	40	Off	70h	70h	0Ch
2	1866	1.35, 1.5	SR	NP	40	Off	70h	70h	0Ch
2	1866	1.35, 1.5	DR	NP	40	Off	70h	70h	0Ch
2	1866	1.5	SR	SR	20	60	75h	75h	0Ch
2	1866	1.5	DR	DR	20	60	75h	75h	0Ch
2	1866	1.5	DR	SR	20	60	75h	75h	0Ch
2	1866	1.5	SR	DR	20	60	75h	75h	0Ch
2	2133	1.5	NP	SR	40	Off	70h	70h	0Ch
2	2133	1.5	NP	DR	40	Off	70h	70h	0Ch
2	2133	1.5	SR	NP	40	Off	70h	70h	0Ch
2	2133	1.5	DR	NP	40	Off	70h	70h	0Ch

1. BIOS programs DrvStrenN = DrvStrenP for each instance.
 BIOS programs instances for “MEMDQSDM” (when used as a data mask) with the same value as DQ.
2. BIOS programs ODTStrenN = ODTStrenP for each instance.
 BIOS programs all used instances with these values.

Table 32: BIOS Recommendations for DDR4 UDIMM Data Bus Configuration

Condition					RTT_Nom	RTT_Wr	RTT_Park	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]0[4,1]_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	EQStrenHiP[6:4], D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
NumDimmSlots	DdrRate	VDDIO	DIMM0	DIMM1						
1	1333	1.2	SR	-	ODT_OFF	RTTWR_OFF	RTTPRK_60	70h	70h	30h
1	1333	1.2	DR	-	ODT_OFF	RTTWR_80	RTTPRK_240	70h	70h	30h
1	1600, 1866, 2133, 2400	1.2	SR	-	ODT_OFF	RTTWR_OFF	RTTPRK_60	7Fh	7Fh	30h
1	1600, 1866, 2133, 2400	1.2	DR	-	ODT_OFF	RTTWR_80	RTTPRK_240	7Fh	7Fh	30h
2	1333	1.2	NP	SR	ODT_OFF	RTTWR_OFF	RTTPRK_60	70h	70h	30h
2	1333	1.2	NP	DR	ODT_OFF	RTTWR_80	RTTPRK_240	70h	70h	30h
2	1333	1.2	SR	NP	ODT_OFF	RTTWR_OFF	RTTPRK_60	70h	70h	30h
2	1333, 1600, 1866	1.2	SR, DR	SR, DR	ODT_40	RTTWR_80	RTTPRK_OFF	75h	75h	30h
2	1333	1.2	DR	NP	ODT_OFF	RTTWR_80	RTTPRK_240	70h	70h	30h
2	1600, 1866, 2133, 2400	1.2	NP	SR	ODT_OFF	RTTWR_OFF	RTTPRK_60	7Fh	7Fh	30h

Table 32: BIOS Recommendations for DDR4 UDIMM Data Bus Configuration (Continued)

Condition					RTT_Nom	RTT_Wr	RTT_Park	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	EQStrenHiP[6:4], D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
NumDimmSlots	DdrRate	VDDIO	DIMM0	DIMM1						
2	1600, 1866, 2133	1.2	NP	DR	ODT_OFF	RTTWR_80	RTTPRK_240	7Fh	7Fh	30h
2	1600, 1866, 2133, 2400	1.2	SR	NP	ODT_OFF	RTTWR_OFF	RTTPRK_60	7Fh	7Fh	30h
2	1600, 1866, 2133	1.2	DR	NP	ODT_OFF	RTTWR_80	RTTPRK_240	7Fh	7Fh	30h
2	2133	1.2	SR	SR	ODT_40	RTTWR_80	RTTPRK_OFF	75h	75h	30h

1. BIOS programs DrvStrenN = DrvStrenP for each instance.
 BIOS programs instances for “MEMDQSDM” (when used as a data mask) with the same value as DQ.
2. BIOS programs ODTStrenP for each instance. BIOS programs ODTStrenN = 0000b.
 BIOS programs all used instances with these values.

Table 33: BIOS Recommendations for DDR4 SODIMM Data Bus Configuration

Condition					RTT_Nom	RTT_Wr	RTT_Park	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	EQStrenHiP[6:4], D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
NumDimmSlots	DdrRate	VDDIO	DIMM0	DIMM1						
1	1333	V1_2	SR	NP	ODT_OFF	RTTWR_OFF	RTTPRK_80	0x70	0x70	0xF
1	1333	V1_2	DR	NP	ODT_OFF	RTTWR_80	RTTPRK_240	0x70	0x70	0x30
1	1600 + 1866 + 2133 + 2400	V1_2	SR	NP	ODT_OFF	RTTWR_OFF	RTTPRK_60	0x7F	0x7F	0x30
1	1600 + 1866 + 2133 + 2400	V1_2	DR	NP	ODT_OFF	RTTWR_80	RTTPRK_240	0x7F	0x7F	0x30
2	1333	V1_2	NP	SR	ODT_OFF	RTTWR_OFF	RTTPRK_80	0x70	0x70	0xF
2	1333	V1_2	NP	DR	ODT_OFF	RTTWR_80	RTTPRK_240	0x70	0x70	0x30
2	1333	V1_2	SR	NP	ODT_OFF	RTTWR_OFF	RTTPRK_80	0x70	0x70	0xF
2	1333 + 1600 + 1866	V1_2	SR + DR	SR + DR	ODT_40	RTTWR_80	RTTPRK_OFF	0x75	0x75	0x30
2	1333	V1_2	DR	NP	ODT_OFF	RTTWR_80	RTTPRK_240	0x70	0x70	0x30
2	1600 + 1866 + 2133	V1_2	NP	SR	ODT_OFF	RTTWR_OFF	RTTPRK_60	0x7F	0x7F	0x30
2	1600 + 1866 + 2133	V1_2	NP	DR	ODT_OFF	RTTWR_80	RTTPRK_240	0x7F	0x7F	0x30

Table 33: BIOS Recommendations for DDR4 SODIMM Data Bus Configuration (Continued)

Condition					RTT_Nom	RTT_Wr	RTT_Park	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQ pins	D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] [DrvStrenP][1] for DQS pins	EQStrenHiP[6:4], D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] [ODTStrenP][2]
NumDimmSlots	DdrRate	VDDIO	DIMM0	DIMM1						
2	1600 + 1866 + 2133	V1_2	SR	NP	ODT_OFF	RTTWR_OFF	RTTPRK_60	0x7F	0x7F	0x30
2	1600 + 1866 + 2133	V1_2	DR	NP	ODT_OFF	RTTWR_80	RTTPRK_240	0x7F	0x7F	0x30
2	2133	V1_2	SR	SR	ODT_40	RTTWR_80	RTTPRK_OFF	0x75	0x75	0x30

1. BIOS programs DrvStrenN = DrvStrenP for each instance.
 BIOS programs instances for "MEMDQSDM" (when used as a data mask) with the same value as DQ.
2. BIOS programs ODTStrenP for each instance. BIOS programs ODTStrenN = 0000b.
 BIOS programs all used instances with these values.

2.9.9.2.6 Phy FIFO Configuration

BIOS programs FIFO configuration values based on NCLK, memory clock, CL, and CWL. A read pointer initial value is specified for each NbPstate as well as one reserved for the PMU during training.

1. For each NbPstate, excluding the "NbPstate PMU" instance, broadcast to all timing groups and chips as follows:
 - D18F2x9C_x00[F,3:0]0_[F,2:0][8,3:0]2E_dct[0][RdPtrInitVal] = Table 34.
 - D18F2x9C_x00[F,8:0]1_0[8,3:0]2E_dct[0][RdPtrInitVal] = Table 34.
2. D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0][RxRdPtrOffset] as follows:
 - Broadcast the value to all chips.

- RxRdPtrOffset = MIN(18, CL).
- See D18F2x22C_dct[0]_mp[1:0][TclAdj].

3. D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0][TxRdPtrOffset] as follows:
 - Broadcast the value to all chips.
 - TxRdPtrOffset = MAX(5, CWL).

Table 34: BIOS Recommendations for DDR FIFO RdPtrInitVal

Condition	NCLK (MHz)								
Condition:DdrRate	500	600	700	800	900	1000	1100	1200	1300
667	14h	14h	14h	14h	14h	14h	14h	14h	14h
800	14h	14h	14h	14h	14h	14h	14h	14h	14h
1066		14h	14h	14h	14h	14h	14h	14h	14h
1333			14h	14h	14h	14h	14h	14h	14h
1600			-	14h	14h	14h	14h	14h	14h
1866			-	-	-	14h	14h	14h	14h
2133			-	-	-	-	10h	10h	10h
2400			-	-	-	-	-	10h	10h

Notes:
1. If exact DdrRate is not shown, use the information from the row of the next highest shown DdrRate.
2. If exact NCLK rate is not shown, use the information from the column of the next highest shown NCLK.
3. Not all conditions are supported. See product definition for details on supported frequencies.

2.9.9.2.7 Phy Predriver Initialization.

Each DDR IO driver has a programmable slew rate controlled by the pre-driver calibration code. The recommended slew rate is a function of the DC drive strength. BIOS initializes the recommended nominal slew rate values as follows:

1. Program D18F2x9C_x0002_0087_dct[0][DisAutoComp, DisPredriverCal] = {1,1}.
2. Program D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0][TxPreN, TxPreP] according to Table 35.
 - For each pad, read D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0][DrvStrenP] and program the corresponding pad in D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0].
 - Use DrvStrenP for memory P-state 0 to determine slew rate.
 - If the chosen DrvStrenP value is not listed in the table for the given DDR rates, then use the next lower DrvStrenP value listed to determine TxPreN and TxPreP.
3. Program D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0][TxPreN, TxPreP] for Cmd/Addr according to Table 36.
 - For each Cmd/Addr pad (MEMCKE[3:0], MEMADD[15:0], MEMBANK[2:0], MEMCS_L[7:0], MEMODT[3:0], MEMCAS_L, MEMWE_L, MEMRAS_L):
 - Read D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0][DrvStrenP] and program the corresponding pad in D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0][TxPreN, TxPreP].
 - Use DrvStrenP for memory P-state 0 to determine slew rate.
 - If the chosen DrvStrenP value is not listed in the table for the given DDR rates, then use the next lower DrvStrenP value listed to determine TxPreN and TxPreP.

4. Program D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0][TxPreN, TxPreP] for clocks according to Table 37.
 - For each clock pad (MEMCLK_H[4:0], MEMCLK_L[4:0]):
 - Read D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0][DrvStrenP] and program the corresponding pad in D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0][TxPreN, TxPreP].
 - Use DrvStrenP for memory P-state 0 to determine slew rate.
 - If the chosen DrvStrenP value is not listed in the table for the given DDR rates, then use the next lower DrvStrenP value listed to determine TxPreN and TxPreP.
5. Program D18F2x9C_x0002_0087_dct[0][DisAutoComp, DisPredriverCal] = {0,0}.

Table 35: Phy Predriver Codes for DDR3 Data/DQS

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.5V	{TxPreP, TxPreN}[2] VDDIO=1.35V	{TxPreP, TxPreN}[2] VDDIO=1.25V
667 - 1067	31h	8Fh	8Fh	8Fh
	70h	8Fh	8Fh	8Fh
	75h	8Dh	8Dh	8Fh
	7Fh	12h	12h	12h
1333 - 1600	31h	8Fh	8Fh	8Fh
	70h	8Fh	8Fh	8Fh
	75h	8Dh	8Dh	8Fh
	7Fh	12h	12h	12h
1866 - 2400	31h	8Fh	8Fh	8Fh
	70h	8Fh	8Fh	8Fh
	75h	8Dh	8Dh	8Fh
	7Fh	12h	12h	23h

1. See D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0].
2. See D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0].

Table 36: Phy Predriver Codes for DDR3 Cmd/Addr

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.5V	{TxPreP, TxPreN}[2] VDDIO=1.35V	{TxPreP, TxPreN}[2] VDDIO=1.25V
667 - 1067	07h	12h	23h	23h
	0Fh	12h	12h	23h
	1Fh	11h	12h	12h
	3Fh	11h	12h	12h
1333 - 1600	07h	12h	23h	23h
	0Fh	12h	12h	23h
	1Fh	11h	12h	23h
	3Fh	11h	12h	23h

Table 36: Phy Predriver Codes for DDR3 Cmd/Addr (Continued)

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.5V	{TxPreP, TxPreN}[2] VDDIO=1.35V	{TxPreP, TxPreN}[2] VDDIO=1.25V
1866 - 2400	07h	23h	23h	34h
	0Fh	22h	23h	34h
	1Fh	22h	22h	34h
	3Fh	22h	22h	34h

1. See D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0].
2. See D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0].

Table 37: Phy Predriver Codes for DDR3 CLK

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.5V	{TxPreP, TxPreN}[2] VDDIO=1.35V	{TxPreP, TxPreN}[2] VDDIO=1.25V
667 - 1067	07h	22h	33h	33h
	0Fh	22h	22h	33h
	1Fh	22h	22h	22h
	3Fh	22h	22h	22h
1333 - 1600	07h	33h	33h	44h
	0Fh	33h	33h	44h
	1Fh	33h	33h	44h
	3Fh	33h	33h	44h
1866 - 2400	07h	44h	56h	67h
	0Fh	45h	56h	67h
	1Fh	55h	56h	67h
	3Fh	55h	56h	66h

1. See D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0].
2. See D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0].

Table 38: Phy Predriver Codes for DDR4 Data/DQS

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.2V
1333 - 2133	31h	8Fh
	70h	8Fh
	75h	8Fh
	7Fh	8Fh

Table 38: Phy Predriver Codes for DDR4 Data/DQS (Continued)

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.2V
2400	31h	8Fh
	70h	8Fh
	75h	8Fh
	7Fh	8Fh
1. See D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0]. 2. See D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0].		

Table 39: Phy Predriver Codes for DDR4 Cmd/Addr

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.2V
1333 - 2133	07h	45h
	0Fh	34h
	1Fh	34h
	3Fh	23h
2400	07h	45h
	0Fh	34h
	1Fh	34h
	3Fh	33h
1. See D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0]. 2. See D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0].		

Table 40: Phy Predriver Codes for DDR4 CLK

DDR Rate	DrvStrenP[1]	{TxPreP, TxPreN}[2] VDDIO=1.2V
1333 - 2133	07h	56h
	0Fh	56h
	1Fh	56h
	3Fh	66h
2400	07h	67h
	0Fh	67h
	1Fh	67h
	3Fh	66h
1. See D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0]. 2. See D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0].		

2.9.9.2.8 Phy Auto-Calibration

BIOS enables a one-time calibration after configuration dependent values are programmed as follows:

1. Initiate one-time calibration. See 2.9.9.2.8.1.
2. After the PMU firmware is loaded, BIOS ensures that the initial calibration is complete prior to executing PMU firmware. See appropriate section for details.
3. After PMU training is complete, or after complete restore of trained values from NVRAM, BIOS initiates auto calibration. See 2.9.9.2.8.3.

2.9.9.2.8.1 One-Time Pre-PMU Calibration

BIOS initiates one calibration. On S5 boot BIOS does this before loading PMU firmware.

1. Program D18F2x9C_x0002_0088_dct[0][CalInitMode, CalOnce] = {1,1}.
2. Program D18F2x9C_x0002_0088_dct[0][CalRun] = 1.
3. Program D18F2x9C_x0002_0088_dct[0][CalOnce,CalRun] = {0,0}.

2.9.9.2.8.2 Fence CalOnce

BIOS ensures the initial one-time calibration is complete as follows:

1. Read D18F2x9C_x0002_0097_dct[0][CalBusy]. This read value is thrown away.
2. Read D18F2x9C_x0002_0097_dct[0][CalBusy] until CalBusy == 0.

2.9.9.2.8.3 Auto Calibration

BIOS initiates auto calibration after PMU training or after complete restore of trained values from NVRAM.

1. Program D18F2x9C_x0002_0088_dct[0][CalInitMode] = 0.
2. Program D18F2x9C_x0002_0088_dct[0][CalRun] = 1.

2.9.9.2.9 PMU Firmware Load

BIOS loads the PMU firmware (see 2.9.8 [PMU]) after each system reset before requesting the PMU to take action with down-stream messages.

Firmware block LENGTH = 16*1024.

1. Program D18F2x9C_x0002_0099_dct[0][PmuReset] = 0 for each phy.
2. Program D18F1x10C[DctCfgBcEn] = 1.
3. For each 16-bit word of the firmware block: Write the word to D18F2x[B,0]9C_x0005_[5FFF:4000]_dct[0] (while using the autoincrement feature).
4. Program D18F1x10C[DctCfgBcEn] = 0.

Additionally, there is a sequence of register writes provided by AMD to write compiler/tool specific data into the data SRAM for use in the PMU executive.

2.9.9.2.10 Phy Registers Required for S3 Resume

To support suspend-to-RAM, BIOS must save registers to NVRAM prior to S3 entry, and restore them when resuming. The following is not a complete list. See 2.9.9 [DCT/DRAM Initialization and Resume] for information on other registers. BIOS restores the following registers, which have been trained by the PMU during boot:

- D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]028_dct[0]
- D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0]
- D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]081_dct[0]
- D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0]
- D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]81_dct[0]
- D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]046_dct[0]
- D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]047_dct[0]

2.9.9.2.11 Calculating Round Trip Command Delays

Software calculates trained round trip delays as follows.

MaxRxCmdDelay, maximum command to data read delay, in units of 1/64 MEMCLK:

- Compute worst case for each chip and when available for each DQ pad within a byte lane.
- MaxRxCmdDelay =
 DQS D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0][RxDly] +
 D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0][RxRdPtrOffset] * 64 +
 DQ (D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0][RxDly] & 0x1F).

2.9.9.3 SPD ROM-Based Configuration

The Serial Presence Detect (SPD) ROM is a non-volatile memory device on the DIMM encoded by the DIMM manufacturer. The description of the SPD is usually provided on a data sheet for the DIMM itself along with data describing the memory devices used. The data describes configuration and speed characteristics of the DIMM and the SDRAM components mounted on the DIMM. The associated data sheet also contains the DIMM byte values that are encoded in the SPD on the DIMM.

BIOS reads the values encoded in the SPD ROM through a system-specific interface. BIOS acquires DIMM configuration information, such as the amount of memory on each DIMM, from the SPD ROM on each DIMM and uses this information to program the DRAM controller registers.

For solder-down DRAM, in the absence of an SPD ROM, BIOS provides the information necessary for DRAM configuration.

The SPD ROM provides values for several DRAM timing parameters that are required by the DCT. In general, BIOS should use the optimal value specified by the SPD ROM.

For Ddr3Mode, these parameters are:

- D18F2x208_dct[0]_mp[1:0][Tref]
- D18F2x200_dct[0]_mp[1:0][Tras]: Active to precharge time
- D18F2x200_dct[0]_mp[1:0][Trp]: Precharge time
- D18F2x200_dct[0]_mp[1:0][Trcd]: RAS to CAS delay
- D18F2x200_dct[0]_mp[1:0][Tcl]: CAS latency
- D18F2x204_dct[0]_mp[1:0][Trtp]: Internal read to precharge command delay time

- D18F2x204_dct[0]_mp[1:0][FourActWindow]: Four activate window delay time
- D18F2x204_dct[0]_mp[1:0][Trrd]: Row active to row active delay
- D18F2x204_dct[0]_mp[1:0][Trc]: Active to active/refresh time
- D18F2x2A0_dct[0]_mp[1:0][Trfc1, Trfc0]: Refresh recovery delay time
- D18F2x2A4_dct[0]_mp[1:0][Trfc3, Trfc2]: Refresh recovery delay time
- D18F2x20C_dct[0]_mp[1:0][Twtr]: Internal write to read command delay time
- D18F2x224_dct[0][Trcpage]: Maximum Activate Count

For Ddr4Mode, these parameters are:

- D18F2x224_dct[0][Trcpage]: SDRAM Optional Features
- D18F2x200_dct[0]_mp[1:0][Tcl]: CAS latency
- D18F2x200_dct[0]_mp[1:0][Trcd]: RAS to CAS Delay Time
- D18F2x200_dct[0]_mp[1:0][Trp]: Row Precharge Delay Time
- D18F2x200_dct[0]_mp[1:0][Tras]: Active to Precharge Delay Time
- D18F2x204_dct[0]_mp[1:0][Trc]: Active to Active/Refresh Delay Time
- D18F2x2A0_dct[0]_mp[1:0][Trfc1, Trfc0]: Refresh Recovery Delay Time
- D18F2x2A4_dct[0]_mp[1:0][Trfc3, Trfc2]: Refresh Recovery Delay Time
- D18F2x204_dct[0]_mp[1:0][FourActWindow]: Four Activate Window Time
- D18F2x204_dct[0]_mp[1:0][Trrd]: Activate to Activate Delay Time, different bank group
- D18F2x204_dct[0]_mp[1:0][TrrdL]: Activate to Activate Delay Time, same bank group
- D18F2x21C_dct[0]_mp[1:0][TrdrdSdScL]: CAS to CAS Delay Time, same bank group
- D18F2x21C_dct[0]_mp[1:0][TwrwrSdScL]: CAS to CAS Delay Time, same bank group
- D18F2x22C_dct[0]_mp[1:0][Twr]: Minimum Write Recovery Time
- D18F2x20C_dct[0]_mp[1:0][Twtr]: Minimum Write to Read Time, different bank group
- D18F2x20C_dct[0]_mp[1:0][TwtrL]: Minimum Write to Read Time, same bank group
- D18F2x1[7C:40]_dct[0]: Connector to SDRAM Bit Mapping

Optimal cycle time is specified for each DIMM and is used to limit or determine bus frequency. See 2.9.9.5 [DRAM Device Initialization and Training].

2.9.9.3.1 DRAM ODT Pin Control

If Ddr3Mode:

BIOS configures the DIMM ODT behavior per chip select according to the DIMM population. In all cases, the processor ODT is off for writes and is on for reads. The ODT pin patterns for reads and writes are programmed using D18F2x[234:230]_dct[0] and D18F2x[23C:238]_dct[0], respectively.

Table 41: DDR3 DIMM ODT Pattern

DIMM0[1]	DIMM1[1]	D18F2x[234:230]_dct[0]		D18F2x[23C:238]_dct[0]	
		D18F2x230	D18F2x234	D18F2x238	D18F2x23C
-	SR	0000_0000h	0000_0000h	0002_0000h	0000_0000h
-	DR	0000_0000h	0000_0000h	0208_0000h	0000_0000h
SR	-	0000_0000h	0000_0000h	0000_0001h	0000_0000h

Table 41: DDR3 DIMM ODT Pattern (Continued)

DIMM0[1]	DIMM1[1]	D18F2x[234:230]_dct[0]		D18F2x[23C:238]_dct[0]	
		D18F2x230	D18F2x234	D18F2x238	D18F2x23C
DR	-	0000_0000h	0000_0000h	0000_0104h	0000_0000h
SR/DR	SR/DR	0101_0202h	0000_0000h	0903_0603h	0000_0000h

1. DIMM0: MEMCSx[1:0], MEMODTx[2,0].
DIMM1: MEMCSx[3:2], MEMODTx[3,1].
Population restrictions may apply. See 2.9.9.2.4 [DRAM Address, Command, and Output Driver Control] for details.

If Ddr4Mode:

Table 42: DDR4 DIMM ODT Pattern

DIMM0[1]	DIMM1[1]	D18F2x[234:230]_dct[0]		D18F2x[23C:238]_dct[0]	
		D18F2x230	D18F2x234	D18F2x238	D18F2x23C
-	SR	0000_0000h	0000_0000h	0000_0000h	0000_0000h
-	DR	0000_0000h	0000_0000h	0000_0000h	0000_0000h
SR	-	0000_0000h	0000_0000h	0000_0000h	0000_0000h
DR	-	0000_0000h	0000_0000h	0000_0000h	0000_0000h
SR	SR	0001_0002h	0000_0000h	0003_0003h	0000_0000h
SR	DR	0101_0002h	0000_0000h	0903_0003h	0000_0000h
DR	SR	0001_0202h	0000_0000h	0003_0603h	0000_0000h
DR	DR	0101_0202h	0000_0000h	0903_0603h	0000_0000h

1. DIMM0: MEMCSx[1:0], MEMODTx[2,0].
DIMM1: MEMCSx[3:2], MEMODTx[3,1].
Population restrictions may apply. See 2.9.9.2.4 [DRAM Address, Command, and Output Driver Control] for details.

2.9.9.4 DCT Specific Configuration

The DCT requires certain features be disabled during DRAM device initialization and training. BIOS should program the registers in Table 43 before DRAM device initialization and training. For normal operation, BIOS programs the recommended values if provided in Table 43. BIOS must quiesce all other forms of DRAM traffic on the channel being trained. See 2.9.9 [DCT/DRAM Initialization and Resume].

Table 43: DCT Training Specific Register Values

Register	Training	Normal Operation
D18F2x78_dct[0][GsyncDis]	0	0
D18F2x218_dct[0]_mp[1:0][TrdrdBan, TrdrdSdSc]	{2h,1h}	See 2.9.9.4.1
D18F2x218_dct[0]_mp[1:0][TrdrdSdDc, TrdrdDd]	{Bh, Bh}	See 2.9.9.4.1
D18F2x214_dct[0]_mp[1:0][TwrwrSdSc]	1	See 2.9.9.4.1
D18F2x214_dct[0]_mp[1:0][TwrwrSdDc, TwrwrDd]	{Bh, Bh}	See 2.9.9.4.1
D18F2x218_dct[0]_mp[1:0][Twrrd]	Bh	See 2.9.9.4.1
D18F2x21C_dct[0]_mp[1:0][TrwtTO]	1Bh	See 2.9.9.4.1
D18F2x21C_dct[0]_mp[1:0][TrdrdSdScL, TwrwrSdScL]	{5h, 5h}	See 2.9.9.4.1

Table 43: DCT Training Specific Register Values (Continued)

Register	Training	Normal Operation
D18F2x78_dct[0][AddrCmdTriEn]	0	1
D18F2x8C_dct[0][DisAutoRefresh]	1	0
D18F2x90_dct[0][ForceAutoPchg]	0	0
D18F2x90_dct[0][DynPageCloseEn]	0	0
D18F2x94_dct[0][BankGroupSwap]	0	0
D18F2x94_dct[0][BankSwizzleMode]	0	1
D18F2x94_dct[0][DcqBypassMax]	0	1Fh
D18F2x94_dct[0][PowerDownEn]	0	1
D18F2x94_dct[0][ZqcsInterval]	00b	10b
D18F2xA4[CmdThrottleMode]	000b	XXXb[1]
D18F2xA4[ODTSEn]	0b	X[1]
D18F2xA4[BwCapEn]	0	X[1]
D18F2xA8_dct[0][BankSwap]	0	1
D18F1x2[1,0][C,4][DctIntLvEn]	0	X[1]
1. Programmed specific to the current platform or memory configuration.		

2.9.9.4.1 DDR Turnaround Parameters

- LD (latency difference) = D18F2x200_dct[0]_mp[1:0][Tcl] - D18F2x20C_dct[0]_mp[1:0][Tcwl].

In all cases, if the computed turn around time is less than the smallest non-reserved value in the register then software programs the smallest non-reserved value.

2.9.9.4.1.1 TrdrdBan, TrdrdSdSc, TrdrdSdDc, and TrdrdDd (Rd->Rd Timing)

The optimal values for D18F2x218_dct[0]_mp[1:0][TrdrdBan, TrdrdSdSc, TrdrdSdDc, TrdrdDd] are platform and configuration specific. After DRAM training, BIOS should use the guidelines below to configure the recommended platform timing values:

- CD_R_R, CD_R_R_SD, and Trdrdban_Phy are obtained from the PMU via the SRAMMsgBlk.
- TrdrdSdSc = 1.
- TrdrdSdDc = CD_R_R_SD.
- TrdrdDd = CD_R_R.
- TrdrdBan = Trdrdban_Phy.

2.9.9.4.1.2 TwrwrSdSc, TwrwrSdDc, TwrwrDd (Wr->Wr Timing)

The optimal values for D18F2x214_dct[0]_mp[1:0][TwrwrSdSc, TwrwrSdDc, TwrwrDd] are platform and configuration specific. After DRAM training, BIOS should use the guidelines below to configure the recommended platform timing values:

- CD_W_W and CD_W_W_SD are obtained from the PMU via the SRAMMsgBlk.
- TwrwrSdSc = 1.
- TwrwrSdDc = CD_W_W_SD.
- TwrwrDd =CD_W_W.

2.9.9.4.1.3 Twrrd (Write to Read DIMM Termination Turn-around)

The optimal value for D18F2x218_dct[0]_mp[1:0][Twrrd] is platform and configuration specific. Prior to DRAM training, BIOS should program this parameter to the largest defined value. After DRAM training, BIOS should use the guidelines below to configure the recommended platform timing values:

- CD_W_R is obtained from the PMU via the SRAMMsgBlk.
- Twrrd = CD_W_R.

2.9.9.4.1.4 TrwtTO (Read-to-Write Turnaround for Data, DQS Contention)

The optimal value for D18F2x21C_dct[0]_mp[1:0][TrwtTO] is platform and configuration specific. Prior to DRAM training, BIOS should program this parameter to the largest defined value. After DRAM training, BIOS should use the guidelines below to configure the recommended platform timing values after DDR training is complete:

- CD_R_W obtained from the PMU via the SRAMMsgBlk.
- TrwtTO = CD_R_W.

2.9.9.5 DRAM Device Initialization and Training

BIOS requests the PMU to initialize the bus and devices, as well as train the processor for optimal operation, using the SequenceCtl bitmap in SRAMMsgBlk, and by taking the PMU out of reset to execute the requests.

In the following steps, BIOS performs each step for each DCT, keeping the PMUs in lockstep before moving to the next step.

1. Perform the PMU firmware load for the module containing Devinit. See 2.9.9.2.9.
2. Program SRAMMsgBlk with all values necessary for Devinit. See 2.9.8.3.
3. Set SRAMMsgBlk, field SequenceCtl = 0x7F for a single module solution, or
 set SRAMMsgBlk, field SequenceCtl as appropriate for a multi-module solution (see below).
4. Fence the CalOnce. See 2.9.9.2.8.2.
5. Program D18F2x9C_x0002_0099_dct[0][PmuStall] = 0.
6. mboxUSPend(PMUQEmpty). See 2.9.8.1.
 - If message is FAIL then continue to step 7 for that DCT only; In keeping with the requirement above, begin next step only after all DCTs are ready.
7. Read SRAMMsgBlk and store for later use.

If the total firmware size makes it necessary to load and execute two firmware modules consecutively for complete boot then BIOS performs the following additional steps. BIOS performs each step for each DCT, keeping the PMUs in lockstep before moving to the next step.

1. Program D18F2x9C_x0002_0099_dct[0][PmuReset,PmuStall] = {1,1}.
2. Program D18F2x9C_x0002_0099_dct[0][PmuReset] = 0.
3. Perform the PMU firmware load for the module containing 2D Read Training. See 2.9.9.2.9.
 - SRAMMsgBlk values from first PMU execution should be preserved. Do not overwrite SRAMMsgBlk.
4. Program SRAMMsgBlk with all values necessary for 2D Read Training. See 2.9.8.3.
5. Set SRAMMsgBlk, field SequenceCtl = 60h.
6. Program D18F2x9C_x0002_0099_dct[0][PmuStall] = 0.

7. mboxUSPend(PMUQEmpty). See 2.9.8.1.

2.9.9.6 DRAM Training

The PMU trains the IO timing and electrical parameters on the DDR bus. BIOS initiates the training using a command bitmap in the SRAMMsgBlk of each PMU, keeping each PMU in lockstep. See 2.9.9.5.

Once the IO timing is trained, the BIOS trains the round trip data latency path from the controller through the channel and back to the northbridge. See 2.9.9.6.1.

2.9.9.6.1 Training MaxRdLatency

After DRAM training, BIOS optimizes D18F2x1F[C:0]_dct[0][MaxRdLatency] using the following algorithm. For MaxRdLatency training, BIOS generates a training pattern using continuous read or write data streams. See 2.9.10.1 [DCT Training Pattern Generation].

For each DCT:
1. Calculate a starting MaxRdLatency delay value by converting the following to NCLKs:
 - D18F2x200_dct[0]_mp[1:0][Tcl] MEMCLKs
 - 2 MEMCLKs
 - 4 NCLKs
 - 18 - (D18F2x9C_x00[F,3:0]0_[F,2:0][8,3:0]2E_dct[0][RdPtrInitVal]/2)) MEMCLKs
2. Select 256 64-byte aligned test addresses associated with the chipselct that has the worst case round trip delay. See 2.9.9.2.11 [Calculating Round Trip Command Delays].
3. Write the DIMM test addresses with the training pattern.
4. For each MaxRdLatency value incrementing from the value calculated in step 1:
 A. Program D18F2x1F[C:0]_dct[0][MaxRdLatency] with the current value.
 B. Read the DIMM test addresses.
 C. Compare the values read against the pattern written.
 - If the pattern is read correctly, go to step 5.
5. Program D18F2x1F[C:0]_dct[0][MaxRdLatency] = CEIL(current value + margin factor).
 - Margin factor = 1 NCLK + 3 MEMCLK.

2.9.9.7 Synchronous Channel Initialization

- BIOS ensures that the frequency is programmed in the DCT correctly prior to the channel initialization. See D18F2x94_dct[0][MemClkFreq] and D18F2x2E0_dct[0][M1MemClkFreq].
- IF (Ddr3Mode) then BIOS programs DCT specific read pointer initial values as specified in step 2 of 2.9.9.2.6 [Phy FIFO Configuration] prior to the channel initialization.
- Program D18F2x78_dct[0][PtrInitReq] = 1.
 Wait for D18F2x78_dct[0][PtrInitReq] == 0.

2.9.9.8 DRAM Channel Disable

The following steps are performed to disable an unused DRAM channel:

1. Program D18F2x9C_x03FF_F041_dct[0] = 0000_0000h.
 - See D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0], and D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0].
2. Program D18F2x9C_x00FA_F04A_dct[0] = 0080h.
 - See D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0].

3. Program D18F2x9C_x0002_000B_dct[0] = 0000_0004h.
4. Ensure that D18F2x78_dct[0][ChanVal] = 0.
5. Ensure that D18F2x90_dct[0][DisDllShutdownSR] = 1.
6. Program D18F2x94_dct[0][DisDramInterface] = 1.

2.9.9.9 DRAM Phy Power Savings

For power savings, BIOS should perform the following actions for each enabled channel:

1. Program D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] and D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] as follows to disable unused pads.
 - DrvStrenN = DrvStrenP = 0.
 - See 2.9.4. Software does this for each unconnected pad in the package or each pad connected to unused pin(s).
2. Program D18F2x9C_x0[3,1:0][F,3:0]0_0014_dct[0][MaxDurDllNoLock] = 0.
3. Program D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] as follows:
 - For M1 context program ODTStrenN = ODTStrenP = 0.
4. Program D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0][PowerDownRcvr] = 1.
5. Program D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0] = 0000h for unused lanes.
6. Disable unused DLL components in ABYTE as follows:
 - Program D18F2x9C_x0020_0077_dct[0] = 0000h.
 - Program D18F2x9C_x0030_0077_dct[0] = 0000h.
 - Program D18F2x9C_x0000_0077_dct[0] = 0004h.
 - Program D18F2x9C_x0010_0077_dct[0] = D18F2x90_dct[0][ParEn] ? 0000h : 0008h.
 - See D18F2x9C_x00[F,3:0]0_0077_dct[0].
7. If ECC memory is not connected or is disabled, or the package does not support ECC, then power down data chiplet 8 with the following:
 - Write to all memory P-state instances if available.
 - Program D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0][PowerDownRcvr] = 1.
 - Program D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0] = 07CFh.
 - Program D18F2x9C_x00[F,8:0]1_0000_dct[0][DByteEnable] = 1.
 - Program D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0] = 07DFh.
 - Program D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] = 0000h.
 - Program D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] = 0000h.
 - Program D18F2x9C_x00[F,8:0]1_[F,B:0]04E_dct[0] = 0000h.
8. Power down unused receivers in data chips as follows:
 - Program D18F2x9C_x00F1_904A_dct[0][PowerDownRcvr] = 1.
 - If x4 DIMMs are not present then program D18F2x9C_x00F1_B04A_dct[0][PowerDownRcvr] = 1.
 - See D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0].
9. Power down the PMU as follows:
 - Program D18F2x9C_x0002_0099_dct[0][PmuReset,PmuStall] = {1,1}.
 - Program D18F2x9C_x0002_0099_dct[0][SRAM_SD] = 1.
 - For M0 context program D18F2x9C_x0[3,1:0]02_0080_dct[0][PMUClkDiv] = 7.
 - D18F2x9C_x0002_005D_dct[0][ForceHClk] = 0.

2.9.10 Continuous Pattern Generation

DRAM training relies on the ability to generate a string of continuous reads or writes between the processor and DRAM, such that worst case electrical interactions can be created. This section describes how these continuous strings of accesses may be generated.

2.9.10.1 DCT Training Pattern Generation

DCT training pattern generation uses pattern generators in the DCT to generate controlled read and write traffic streams. During write pattern generation, data values based off of a deterministic pattern are burst to the DRAM interface. Conversely for reads, data bursts from the DRAM interface are compared against expected data values on a per nibble basis.

Two address modes are available for DRAM training pattern generation, as configured by D18F2x250_dct[0][CmdTgt]. For generating a stream of reads or writes to the same rank, address target A mode is used. To generate a stream of accesses to up to two different ranks, address target A and B mode is used.

An overview of the BIOS sequence to generate training patterns is as follows:

- Configure the DCT for pattern generation. See 2.9.9.4 [DCT Specific Configuration].
- Ensure DIMMs are configured to support 8-beat bursts (BL8 or dynamic burst length on the fly).
- Wait for D18F2x250_dct[0][CmdSendInProg] == 0.
- Program D18F2x250_dct[0][CmdTestEnable] = 1.
- Send activate commands as appropriate. See 2.9.10.1.1 [Activate and Precharge Command Generation].
- Send read or write commands as desired. See 2.9.10.1.2 [Read and Write Command Generation].
- Send precharge commands as appropriate. See 2.9.10.1.1 [Activate and Precharge Command Generation].
- Program D18F2x250_dct[0][CmdTestEnable] = 0.

2.9.10.1.1 Activate and Precharge Command Generation

Prior to sending read or write commands, BIOS must send an activate command to a row in a particular bank of the DRAM devices for access. To send an activate command, BIOS performs the following steps:

- Program D18F2x28C_dct[0] to the desired address as follows:
 - CmdChipSelect = CS[7:0].
 - CmdBank = BA[2:0].
 - CmdAddress = A[17:0].
- Program D18F2x28C_dct[0][SendActCmd] = 1.
- Wait until D18F2x28C_dct[0][SendActCmd] == 0.
- Wait 75 MEMCLKs.

After completing its accesses, BIOS must deactivate open rows in the DRAM devices. To send a precharge or precharge all command to deactivate open rows in a bank or in all banks, BIOS performs the following steps:

- Wait 25 MEMCLKs.
- Program D18F2x28C_dct[0] to the desired address as follows:
 - CmdChipSelect = CS[7:0].
 - Precharge all command:
 - CmdAddress[10] = 1.
 - Precharge command:
 - CmdAddress[10] = 0.
 - CmdBank = BA[2:0].

- Program D18F2x28C_dct[0][SendPchgCmd] = 1.
- Wait until D18F2x28C_dct[0][SendPchgCmd] == 0.
- Wait 25 MEMCLKs.

2.9.10.1.2 Read and Write Command Generation

BIOS performs the following steps for read pattern generation:

- Program D18F2x278_dct[0] and D18F2x274_dct[0] with the data comparison masks for bit lanes of interest.
- Program D18F2x270_dct[0][DataPrbsSeed] the seed for the desired PRBS.
- Program D18F2x260_dct[0][CmdCount] equal to the number of cache line commands.
- Program D18F2x25C_dct[0][BubbleCnt, CmdStreamLen]. See 2.9.10.1.5 [BubbleCnt and CmdStreamLen Programming].
- Program D18F2x25[8,4]_dct[0] to the initial address.
- Program D18F2x250_dct[0] as follows:
 - ResetAllErr and StopOnErr as desired. See 2.9.10.1.4 [Data Comparison].
 - CmdTgt corresponding to D18F2x25[8,4]_dct[0].
 - CmdType = 000b.
 - SendCmd = 1.
- If D18F2x260_dct[0][CmdCount] != 0 then
 Wait for D18F2x250_dct[0][TestStatus] == 1 and D18F2x250_dct[0][CmdSendInProg] == 0.
 Else
 Wait the desired amount of time.
 Program D18F2x260_dct[0][CmdCount] = 1.
 Wait for D18F2x250_dct[0][TestStatus] == 1 and D18F2x250_dct[0][CmdSendInProg] == 0.
- Program D18F2x250_dct[0][SendCmd] = 0.
- Read D18F2x264_dct[0], D18F2x268_dct[0], and D18F2x26C_dct[0] if applicable.

BIOS performs the following steps for write pattern generation:

- Program D18F2x270_dct[0][DataPrbsSeed] the seed for the desired PRBS.
- Program D18F2x260_dct[0][CmdCount] equal to the number of cache line commands desired.
- Program D18F2x25C_dct[0][BubbleCnt, CmdStreamLen]. See 2.9.10.1.5 [BubbleCnt and CmdStreamLen Programming].
- Program D18F2x25[8,4]_dct[0] to the initial address.
- Program D18F2x250_dct[0] as follows:
 - CmdTgt corresponding to D18F2x25[8,4]_dct[0].
 - CmdType = 001b.
 - SendCmd = 1.
- If D18F2x260_dct[0][CmdCount] != 0 then
 Wait for D18F2x250_dct[0][TestStatus] == 1 and D18F2x250_dct[0][CmdSendInProg] == 0.
 Else
 Wait the desired amount of time.
 Program D18F2x260_dct[0][CmdCount] = 1.
 Wait for D18F2x250_dct[0][TestStatus] == 1 and D18F2x250_dct[0][CmdSendInProg] == 0.
- Program D18F2x250_dct[0][SendCmd] = 0.

BIOS combines the two sets of steps listed above for alternating write and read pattern generation.

2.9.10.1.3 Configurable Data Patterns

In addition to PRBS mode, D18F2x250_dct[0][DataPatGenSel] and D18F2x2[B4,B0,AC,A8]_dct[0] allow configurable data pattern generation.

Table 44. Configurable Data Pattern Example with 1 Address Target

DQ/ Beat	Write TgtA Cmd 0								Write TgtA Cmd 1								Write TgtA Cmd 2								...
	Beat 0	Beat 1	Beat 2	Beat 3	Beat 4	Beat 5	Beat 6	Beat 7	Beat 8	Beat 9	Beat 10	Beat 11	Beat 12	Beat 13	Beat 14	Beat 15	Beat 16	Beat 17	Beat 18	Beat 19	Beat 20	Beat 21	Beat 22	Beat 23	...
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	
2	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	
3	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	
4	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	
5	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	
6	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	
7	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	
8	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	
9	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	
10	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	
11	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	
12	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	
13	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	
14	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
...																									

1. D18F2x250_dct[0][DataPatGenSel] = 10b.
 D18F2x2[B4,B0,AC,A8]_dct[0] = {FFEE_DDCCh, BBAA_9988h, 7766_5544h, 3322_1100h}.

Table 45. Configurable Data Pattern Circular Shift Example with 1 Address Target

DQ/ Beat	Write TgtA Cmd 0								Write TgtA Cmd 1								Write TgtA Cmd 2								...
	Beat 0	Beat 1	Beat 2	Beat 3	Beat 4	Beat 5	Beat 6	Beat 7	Beat 8	Beat 9	Beat 10	Beat 11	Beat 12	Beat 13	Beat 14	Beat 15	Beat 16	Beat 17	Beat 18	Beat 19	Beat 20	Beat 21	Beat 22	Beat 23	...
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	
1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	
2	0	1	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	
3	1	1	0	0	1	1	0	0	0	1	0	0	0	1	0	0	1	0	0	0	1	0	0	0	
4	0	0	1	0	0	0	1	0	1	1	0	0	1	1	0	0	0	1	0	0	0	1	0	0	
5	1	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	1	1	0	0	1	1	0	0	
6	0	1	1	0	0	1	1	0	1	0	1	0	1	0	1	0	0	0	1	0	0	0	1	0	
7	1	1	1	0	1	1	1	0	0	1	1	0	0	1	1	0	1	0	1	0	1	0	1	0	
8	0	0	0	1	0	0	0	1	1	1	1	0	1	1	1	0	0	1	1	0	0	1	1	0	
9	1	0	0	1	1	0	0	1	0	0	0	1	0	0	0	1	1	1	1	0	1	1	1	0	
10	0	1	0	1	0	1	0	1	1	0	0	1	1	0	0	1	0	0	0	1	0	0	0	1	
11	1	1	0	1	1	1	0	1	0	1	0	1	0	1	0	1	1	0	0	1	1	0	0	1	
12	0	0	1	1	0	0	1	1	1	1	0	1	1	1	0	1	0	1	0	1	0	1	0	1	
13	1	0	1	1	1	0	1	1	0	0	1	1	0	0	1	1	1	1	0	1	1	1	0	1	
14	0	1	1	1	0	1	1	1	1	0	1	1	1	0	1	1	0	0	1	1	0	0	1	1	
15	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	0	1	1	1	0	1	1	
...																									

1. D18F2x250_dct[0][DataPatGenSel] = 11b.
 D18F2x2[B4,B0,AC,A8]_dct[0] = {FFEE_DDCCh, BBAA_9988h, 7766_5544h, 3322_1100h}.

2.9.10.1.3.1 Static Data Pattern Override

Software is also able to use the bits of D18F2x280_dct[0], D18F2x284_dct[0], and D18F2x288_dct[0] to override the DCT pattern generator on a DQ basis. This can only be used in the D18F2x250_dct[0][DataPatGenSel] == 10b mode. Software programs the following to enable the mode:

- Program D18F2x288_dct[0][PatOvrVal] to the data value desired.
- Program D18F2x280_dct[0][DQPatOvrEn[31:0]] and D18F2x284_dct[0][DQPatOvrEn[63:32]] to enable the override on the desired bit lanes.

Table 46. Data Pattern Override Example with 1 Address Target

DQ/ Beat	Write TgtA Cmd 0								Write TgtA Cmd 1								Write TgtA Cmd 2								...
	Beat 0	Beat 1	Beat 2	Beat 3	Beat 4	Beat 5	Beat 6	Beat 7	Beat 8	Beat 9	Beat 10	Beat 11	Beat 12	Beat 13	Beat 14	Beat 15	Beat 16	Beat 17	Beat 18	Beat 19	Beat 20	Beat 21	Beat 22	Beat 23	...
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	
2	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	
3	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	
4	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	
5	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	

Table 46. Data Pattern Override Example with 1 Address Target (Continued)

6	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	
9	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	
10	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	
11	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	
12	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	
13	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	
14	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
...																									

1. D18F2x250_dct[0][DataPatGenSel] = 10b.
 D18F2x2[B4,B0,AC,A8]_dct[0] = {FFEE_DDCCh, BBAA_9988h, 7766_5544h, 3322_1100h}.
 D18F2x288_dct[0][PatOvrVal] = 0.
 D18F2x280_dct[0][DQPatOvrEn[31:0]] = 0x0000_0080.
 D18F2x284_dct[0][DQPatOvrEn[63:32]] = 0x0000_0000.

2.9.10.1.3.2 Xor Data Pattern Override

Software is also able to use the bits of D18F2x288_dct[0][XorPatOvr] to override the DCT pattern generator with the same XOR function for all byte lanes. This feature can only be used with select D18F2x250_dct[0][DataPatGenSel] PRBS modes. Software programs the following to enable the mode:

- Program D18F2x288_dct[0][XorPatOvr] to the data value of the XOR function desired.
 - Output data = IF (DQPatOvrEn) THEN PatOvrVal. ELSE (DataPatGen ^ XorPatOvr). ENDIF.

2.9.10.1.4 Data Comparison

The DCT compares the incoming read data against the expected pattern sequence during pattern generation. BIOS may choose to continue command generation and accumulate errors or stop command generation on the first error occurrence by programming D18F2x250_dct[0][StopOnErr].

Error information is reported via D18F2x264_dct[0], D18F2x268_dct[0], D18F2x26C_dct[0], D18F2x294_dct[0], and D18F2x298_dct[0]. Error information can be masked on per-bit basis by programming D18F2x274_dct[0], and D18F2x278_dct[0].

BIOS resets the error information by programming D18F2x250_dct[0][ResetAllErr] = 1.

Error information is only valid in certain modes of D18F2x250_dct[0][CmdType, CmdTgt] and D18F2x260_dct[0][CmdCount] and when using 64-byte aligned addresses in D18F2x25[8,4]_dct[0][TgtAddress]. Some modes require a series of writes to setup a DRAM data pattern. See Table 47.

Table 47. Command Generation and Data Comparison

Commands	CmdType	Cmd Tgt	Maximum CmdCount[4]
Read	000b	00b[1]	128
		01b[1]	256[2]
Write-Read	010b	00b	Infinite
		01b[3]	256[2]

1. Requires setup writes to store a data pattern in DRAM. The write commands are generated using the same CmdTgt, CmdCount, and DataPrbsSeed settings.
2. D18F2x254[TgtAddress] != D18F2x258[TgtAddress].
3. Requires setup writes to store a data pattern in DRAM. The write commands are generated programming D18F2x254[TgtAddress] to the intended Target B, CmdTgt = 00b, CmdCount to 1/2 of the intended command count, and the same DataPrbsSeed setting.
4. D18F2x250_dct[0][LsfrRollOver] = 0. The maximum CmdCount is infinite for all modes listed if D18F2x250_dct[0][LsfrRollOver] == 1.

2.9.10.1.4.1 Activate and Precharge Traffic Generation

The DCT generates ACT and PRE traffic in the available command bandwith during a WR or RD sequence generated by D18F2x250_dct[0][SendCmd] when D18F2x250_dct[0][ActPchgGenEn] == 1. This 16 command sequence repeats until the WR or RD initiated by SendCmd hits a stopping condition.

Software is able to adjust the command spacing and address sequence by programming D18F2x28C_dct[0][CmdChipSelect, CmdAddress[17:4]], D18F2x2B8_dct[0][ActPchgSeq, ActPchgCmdMin], and D18F2x2[C0,BC]_dct[0].

The banks specified by D18F2x2[C0,BC]_dct[0] must be in the idle state and must not conflict with the banks used by the SendCmd WR or RD traffic stream.Software is responsible for the ACT/PRE protocol of the WR or RD targets initiated by SendCmd. On a stopping condition for the WR or RD traffic stream, software is responsible for returning all of the possibly open banks to the idle state.

- Example configuration for DRAM Training:
- D18F2x250_dct[0][ActPchgGenEn, CmdTgt, CmdType] = {1b, 1b, 0b}. // Tgt A&B, Reads
- D18F2x28C_dct[0][CmdChipSelect] = CS of TgtA.
- D18F2x28C_dct[0][CmdAddress[17:4]] = desired upper row address bits.
 - Hardware places the sequence number in row address [3:0].
- D18F2x2B8_dct[0][ActPchgSeq, ActPchgCmdMin] = {0F0Fh, Fh}.
- D18F2x2BC_dct[0] = 5432_ 5432h. // Banks 2,3,4,5
- D18F2x2C0_dct[0] = 7632_ 7632h. // Banks 2,3,6,7

2.9.10.1.5 BubbleCnt and CmdStreamLen Programming

BIOS programs D18F2x25C_dct[0][BubbleCnt2, BubbleCnt, CmdStreamLen] to ensure proper channel command spacing in command generation mode.

For continuous pattern generation it is expected that BubbleCnt = 0. In other modes, BIOS programs BubbleCnt, BubbleCnt2, and CmdStreamLen greater than or equal to the relevant DRAM timing parameters shown below to prevent contention on the DRAM bus. In all cases, if the minimum BubbleCnt > 0 or CmdType == 010b then BIOS programs CmdStreamLen = 1.

- Software should also ensure bubble programming satisfies D18F2x21C_dct[0]_mp[1:0][TrdrdSdScL, TwrwrSdScL].

Table 48. Command Generation and BubbleCnt Programming

Commands	CmdType	CmdTgt	BubbleCnt	BubbleCnt2
Read-Read same CS	000b	0Xb	D18F2x218_dct[0]_mp[1:0][TrdrdSdSc] - 1; Exclude banned spacing: D18F2x218_dct[0]_mp[1:0][TrdrdBan]	Xh
Write-Write same CS	001b	0Xb	D18F2x214_dct[0]_mp[1:0][TwrwrSdSc] - 1	Xh
Write-Read same CS	010b	00b	D18F2x20C_dct[0]_mp[1:0][Twtr] + D18F2x20C_dct[0]_mp[1:0][Tcwl] + 4 - 1	D18F2x21C_dct[0]_mp[1:0][TrwtTO] - 1
Read-Read different CS	000b	01b	D18F2x218_dct[0]_mp[1:0][TrdrdSdDc] - 1; Exclude banned spacing: D18F2x218_dct[0]_mp[1:0][TrdrdBan]	Xh
Write-Write different CS	001b	01b	D18F2x214_dct[0]_mp[1:0][TwrwrSdDc] - 1	Xh
Write-Read different CS	010b	01b	D18F2x218_dct[0]_mp[1:0][Twrrd] - 1	D18F2x21C_dct[0]_mp[1:0][TrwtTO] - 1
Read-Read different DIMM	000b	01b	D18F2x218_dct[0]_mp[1:0][TrdrdDd] - 1; Exclude banned spacing: D18F2x218_dct[0]_mp[1:0][TrdrdBan]	Xh
Write-Write different DIMM	001b	01b	D18F2x214_dct[0]_mp[1:0][TwrwrDd] - 1	Xh
Write-Read different DIMM	010b	01b	D18F2x218_dct[0]_mp[1:0][Twrrd] - 1	D18F2x21C_dct[0]_mp[1:0][TrwtTO] - 1

2.9.11 Memory Interleaving Modes

The processor supports the following memory interleaving modes:

- Chip select: interleaves the physical address space over multiple DIMM ranks on a channel, as opposed to each DIMM owning single consecutive address spaces. See 2.9.11.1 [Chip Select Interleaving].

Table 49. Recommended Interleave Configurations

Interleaving Mode	Enabled	Disabled
Chip Select Interleaving	Number of chip selects installed on the channel is a power of two.	Requirements not satisfied.

2.9.11.1 Chip Select Interleaving

The chip select memory interleaving mode has the following requirements:

- The number of chip selects interleaved is a power of two.
- The chip selects are the same size and type.

A BIOS algorithm for programming D18F2x[5C:40]_dct[0] [DRAM CS Base Address] and D18F2x[6C:60]_dct[0] [DRAM CS Mask] in memory interleaving mode is as follows:

1. Program all DRAM CS Base Address and DRAM CS Mask registers using contiguous normalized address mapping.
2. For each enabled chip select, swap the corresponding BaseAddr[38:27] bits with the BaseAddr[21:11] bits as defined in Table 50.
3. For each enabled chip select, swap the corresponding AddrMask[38:27] bits with the AddrMask[21:11] bits as defined in Table 50.

Table 50. DDR3 Swapped Normalized Address Lines for CS Interleaving

Condition				Swapped Base Address and Address Mask bits	
DIMM Address Map[1]	CS Size	BankSwap[2]	BankSwapLoAddress[2]	4-way CS interleaving	2-way CS interleaving
0001b	256MB	0, 1	-	[29:28] and [17:16]	[28] and [16]
		1	8	[29:28] and [12:11]	[28] and [11]
		1	9	[29:28] and [13:12]	[28] and [12]
0010b	512MB	0, 1	-	[30:29] and [17:16]	[29] and [16]
		1	8	[30:29] and [12:11]	[29] and [11]
		1	9	[30:29] and [13:12]	[29] and [12]
0101b	1GB	0, 1	-	[31:30] and [17:16]	[30] and [16]
		1	8	[31:30] and [12:11]	[30] and [11]
		1	9	[31:30] and [13:12]	[30] and [12]

Table 50. DDR3 Swapped Normalized Address Lines for CS Interleaving (Continued)

Condition				Swapped Base Address and Address Mask bits	
DIMM Address Map[1]	CS Size	BankSwap[2]	BankSwapLoAddress[2]	4-way CS interleaving	2-way CS interleaving
0110b	1GB	0, 1	-	[31:30] and [18:17]	[30] and [17]
		1	8	[31:30] and [12:11]	[30] and [11]
		1	9	[31:30] and [13:12]	[30] and [12]
0111b	2GB	0, 1	-	[32:31] and [17:16]	[31] and [16]
		1	8	[32:31] and [12:11]	[31] and [11]
		1	9	[32:31] and [13:12]	[31] and [12]
1000b	2GB	0, 1	-	[32:31] and [18:17]	[31] and [17]
		1	8	[32:31] and [12:11]	[31] and [11]
		1	9	[32:31] and [13:12]	[31] and [12]
1001b	4GB	0, 1	-	[33:32] and [18:17]	[32] and [17]
		1	8	[33:32] and [12:11]	[32] and [11]
		1	9	[33:32] and [13:12]	[32] and [12]
1010b	4GB	0, 1	-	[33:32] and [17:16]	[32] and [16]
		1	8	[33:32] and [12:11]	[32] and [11]
		1	9	[33:32] and [13:12]	[32] and [12]
1011b	8GB	0, 1	-	[34:33] and [18:17]	[33] and [17]
		1	8	[34:33] and [12:11]	[33] and [11]
		1	9	[34:33] and [13:12]	[33] and [12]
1100b	16GB	0, 1	-	[35:34] and [19:18]	[34] and [18]
		1	8	[35:34] and [12:11]	[34] and [11]
		1	9	[35:34] and [13:12]	[34] and [12]

1. See D18F2x80_dct[0] [DRAM Bank Address Mapping].
2. See D18F2xA8_dct[0][BankSwap] and D18F2x110[BankSwapAddr8En, DctSelIntLvAddr]. IF (BankSwapAddr8En == 1 && DctSelIntLvAddr == 100b) THEN BankSwapLoAddress = 8. ELSE BankSwapLoAddress = 9. ENDIF.

Table 51. DDR4 Swapped Normalized Address Lines for CS Interleaving

Condition				Swapped Base Address and Address Mask bits	
DIMM Address Map[1]	CS Size	BankSwap[2]	BankSwapLoAddress[2]	4-way CS interleaving	2-way CS interleaving
1h	1GB	0, 1	-	[31:30] and [17:16]	[30] and [16]
		1	8	[31:30] and [12:11]	[30] and [11]
		1	9	[31:30] and [13:12]	[30] and [12]
2h	2GB	0, 1	-	[32:31] and [18:17]	[31] and [17]
		1	8	[32:31] and [13:12]	[31] and [12]
		1	9	[32:31] and [14:13]	[31] and [13]
3h	2GB	0, 1	-	[32:31] and [17:16]	[31] and [16]
		1	8	[32:31] and [12:11]	[31] and [11]
		1	9	[32:31] and [13:12]	[31] and [12]
4h	4GB	0, 1	-	[33:32] and [18:17]	[32] and [17]
		1	8	[33:32] and [13:12]	[32] and [12]
		1	9	[33:32] and [14:13]	[32] and [13]
5h	4GB	0, 1	-	[33:32] and [17:16]	[32] and [16]
		1	8	[33:32] and [12:11]	[32] and [11]
		1	9	[33:32] and [13:12]	[32] and [12]
6h	8GB	0, 1	-	[34:33] and [18:17]	[33] and [17]
		1	8	[34:33] and [13:12]	[33] and [12]
		1	9	[34:33] and [14:13]	[33] and [13]
7h	8GB	0, 1	-	[34:33] and [17:16]	[33] and [16]
		1	8	[34:33] and [12:11]	[33] and [11]
		1	9	[34:33] and [13:12]	[33] and [12]
8h	16GB	0, 1	-	[35:34] and [18:17]	[34] and [17]
		1	8	[35:34] and [13:12]	[34] and [12]
		1	9	[35:34] and [14:13]	[34] and [13]

Table 51. DDR4 Swapped Normalized Address Lines for CS Interleaving (Continued)

Condition				Swapped Base Address and Address Mask bits	
DIMM Address Map[1]	CS Size	BankSwap[2]	BankSwapLoAddress[2]	4-way CS interleaving	2-way CS interleaving
Ah	32GB	0, 1	-	[36:35] and [18:17]	[35] and [17]
		1	8	[36:35] and [13:12]	[35] and [12]
		1	9	[36:35] and [14:13]	[35] and [13]

1. See D18F2x80_dct[0] [DRAM Bank Address Mapping].
2. See D18F2xA8_dct[0][BankSwap] and D18F2x110[BankSwapAddr8En, DctSelIntLvAddr]. IF (BankSwapAddr8En == 1 && DctSelBankSwap == 1) THEN BankSwapLoAddress = 8. ELSE BankSwapLoAddress = 9. ENDIF.

2.9.12 Memory Hoisting

Memory hoisting reclaims the otherwise inaccessible DRAM that would naturally reside in memory regions used by MMIO. When memory hoisting is configured by BIOS, DRAM physical addresses are repositioned above the 4 GB address level in the address map. In operation, the physical addresses are remapped in hardware to the normalized addresses used by a DCT.

The region of DRAM that is hoisted is defined to be from D18F1xF0[DramHoleBase] to the 4 GB level. Hoisting is enabled by programming D18F1xF0 [DRAM Hole Address] and configuring the DCTs per the equations in this section.

DramHoleSize is defined in order to simplify the following equations in this section and is calculated as follows:

- Define the DRAM hole region as DramHoleSize[31:24] = 100h - D18F1xF0[DramHoleBase[31:24]].

2.9.12.1 DramHoleOffset Programming

D18F1xF0[DramHoleOffset] is programmed to account for the addresses from D18F1xF0[DramHoleBase] to 4 GB when it falls inside of a D18F1x2[1,0][8,0][DctBaseAddr] and D18F1x2[1,0][C,4][DctLimitAddr] region. See Figure 3 as an example memory population.

- Program D18F1xF0[DramHoleOffset[31:23]] = {DramHoleSize[31:24], 0b} + {DctBaseAddr[31:27], 0000b}.

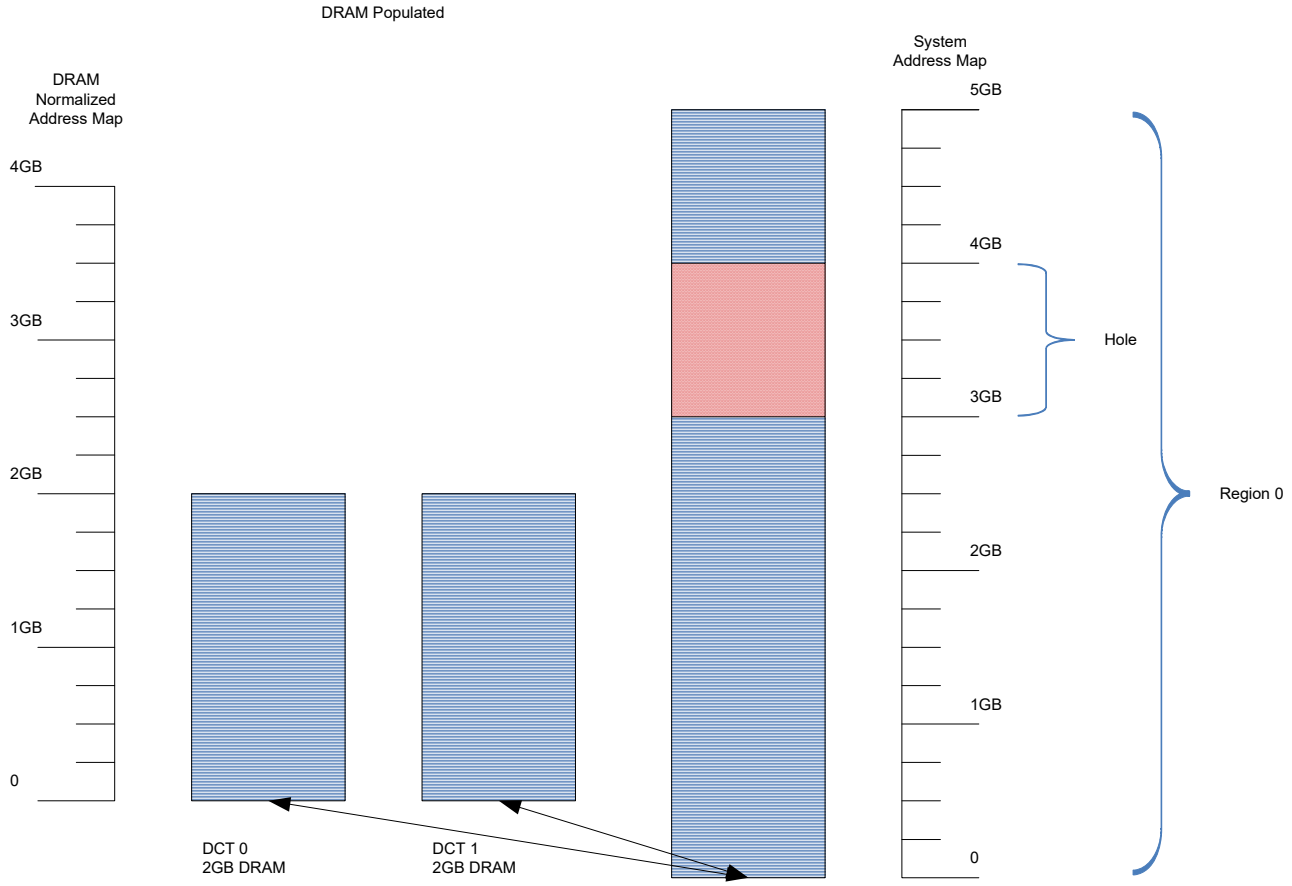


Figure 3: Memory Configuration with Memory Hole Inside of Region

D18F1xF0[DramHoleOffset] is unused when the memory hole falls outside of a region. Figure 4 shows an example memory population which uses two memory regions. Region 1 is configured to begin above the memory hole.

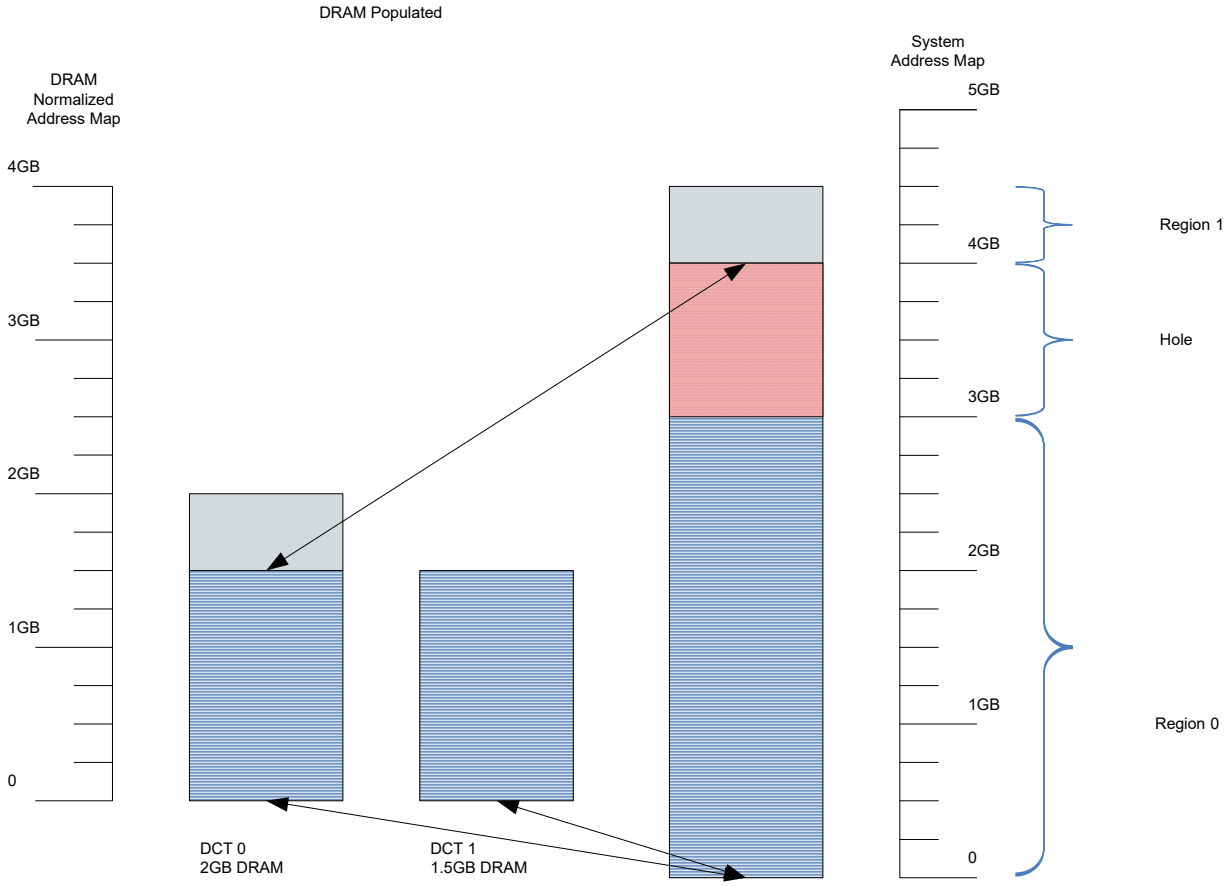


Figure 4: Memory Configuration with Memory Hole Outside of Region

2.9.13 DRAM CC6/PC6 Storage

DRAM is used to hold the state information of cores entering the CC6 power management state. As part of the system setup if CC6 or PC6 is enabled, BIOS configures a special region of DRAM to hold the state information. In operation, hardware protects this region from general system accesses while allowing the cores access during C-state transitions.

2.9.13.1 Fixed Storage

The size of each special DRAM storage region is defined to be a fixed 16MB. BIOS configures the storage region at the top of the DRAM range, adjusts D18F1x[7:4][C,4][DramLimit] and the processor top of DRAM specified by MSRC001_001A[TOM] or MSRC001_001D[TOM2] downward accordingly. See Table 52.

After finalizing the system DRAM configuration, BIOS must set D18F2x118[LockDramCfg] = 1 to enable the hardware protection.

Table 52. Example storage region configuration

Node	DRAM Populated	D18F1x[17C:140,7C:40] [DramBase, DramLimit]	CC6 DRAM Range	D18F4x128 [CoreStateSaveDestNode]	D18F1x120[DramBaseAddr], D18F1x124[DramLimitAddr]
0	256 MB	0 MB, 240 MB - 1	240 MB, 256 MB - 1	0	0 MB, 256 MB - 1

2.9.14 DRAM On DIMM Thermal Management and Power Capping

Each DCT can throttle commands based on the state of the channel EVENT_L pin or when D18F2xA4[BwCapEn] == 1. The EVENT_L pin is used for thermal management while D18F2xA4[BwCapEn] limits memory power independent of the thermal management solution.

The EVENT_L pin for each channel must be wire OR'ed. If all DCTs enabled throttle commands in lockstep using the amount specified in D18F2xA4[CmdThrottleMode] and D18F2xA4[BwCapCmdThrottleMode].

The recommended BIOS configuration for the EVENT_L pin is as follows:

- BIOS may enable command throttling on a DRAM controller if the platform supports the EVENT_L pin by programming D18F2xA4[ODTSEn] = 1.
 - The recommended usage is for this pin to be connected to one or more JEDEC defined on DIMM temperature sensors. The DIMM SPD ROM indicates on DIMM temperature sensor support.
 - BIOS configures the temperature sensor(s) to assert EVENT_L pin active low when the trip point is exceeded and deassert EVENT_L when the temperature drops below the trip point minus the sensor defined hysteresis.
 - BIOS programs D18F2xA4[CmdThrottleMode] with the throttling mode to employ when the trip point has been exceeded.
 - The hardware enforces a refresh rate of 3.9 us while EVENT_L is asserted.
- BIOS configures D18F2x208_dct[0]_mp[1:0] based on JEDEC defined temperature range options, as indicated by the DIMM SPD ROM. The two defined temperature ranges are normal (with a case temperature of 85 °C) and extended (with a case temperature of 95 °C).
 - If all DIMMs support the normal temperature range, or if normal and extended temperature range DIMMs are mixed, BIOS programs Tref to 7.8 us and D18F2xA4[ODTSEn] = 1. BIOS configures the temperature sensor trip point for all DIMMs according to the 85 °C case temperature specification.
 - If all DIMMs support the extended temperature range, BIOS has two options:
 a. Follow the recommendation for normal temperature range DIMMs.
 b. Program Tref = 3.9 us and configure the temperature sensor trip point for all DIMMs according to the 95 °C case temperature specification.
- At startup, the BIOS determines if the DRAMs are hot before enabling a DCT and delays for an amount of time to allow the devices to cool under the influence of the thermal solution. This is accomplished by checking the temperature status in the temperature sensor of each DIMM.
- The DCT latched status of the EVENT_L pin for can be read by system software in D18F2xAC [DRAM Controller Temperature Status].

The relationship between the DRAM case temperature, trip point, and EVENT_L pin sampling interval is outlined as follows:

- The trip point for each DIMM is ordinarily configured to the case temperature specification minus a guardband temperature for the DIMM.
- The temperature guardband is vendor defined and is used to account for sensor inaccuracy, EVENT_L pin

sample interval, and platform thermal design.
- The sampling interval is vendor defined. It is expected to be approximately 1 second.

BIOS may enable bandwidth capping on a DRAM controller by setting D18F2xA4[BwCapEn] = 1 and programming D18F2xA4[BwCapCmdThrottleMode] with the throttling mode to employ. The DCT will employ the larger of the two throttling percentages as specified by D18F2xA4[BwCapCmdThrottleMode] and D18F2xA4[CmdThrottleMode] if the EVENT_L pin is asserted when both D18F2xA4[BwCapEn] == 1 and D18F2xA4[ODTSEn] == 1.

2.10 Thermal Functions

Thermal functions HTC, PROCHOT_L and THERMTRIP are intended to maintain processor temperature in a valid range by:

- Providing a signal to external circuitry for system thermal management like fan control.
- Lowering power consumption by switching to lower-performance P-state.
- Sending processor to the THERMTRIP state to prevent it from damage.

The processor thermal-related circuitry includes (1) the temperature calculation circuit (TCC) for determining the temperature of the processor and (2) logic that uses the temperature from the TCC.

2.10.1 The T_{ctl} Temperature Scale

T_{ctl} is a processor temperature control value used for processor thermal management. T_{ctl} is accessible through D0F0xBC_xD820_0CA4[CUR_TEMP]. T_{ctl} is a temperature on its own scale aligned to the processors cooling requirements. Therefore T_{ctl} does not represent a temperature which could be measured on the die or the case of the processor. Instead, it specifies the processor temperature relative to the maximum operating temperature, T_{ctl_max}. T_{ctl_max} is specified in the power and thermal data sheet. T_{ctl} is defined as follows for all parts:

A: For $T_{ctl} = T_{ctl_max}$ to 255.875: the temperature of the part is $[T_{ctl} - T_{ctl_max}]$ over the maximum operating temperature. The processor may take corrective actions that affect performance, such as HTC, to support the return to T_{ctl} range B.

B: For $T_{ctl} = 0$ to $T_{ctl_max} - 0.125$: the temperature of the part is $[T_{ctl_max} - T_{ctl}]$ under the maximum operating temperature.

It is recommended that the external cooling solution reach maximum capability (such as fan speed) at least 6C degrees below T_{ctl_max}.

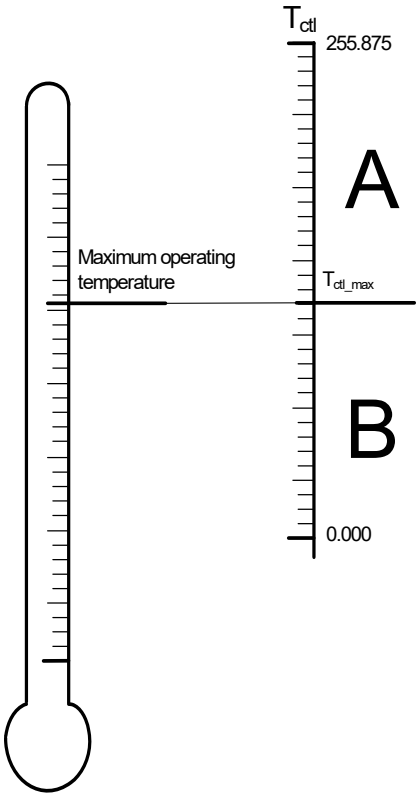


Figure 5: T_{ctl} Scale

2.10.2 Temperature Slew Rate Control

The temperature slew rate controls in D0F0xBC_xD820_0CA4 are used to filter the processor temperature provided in D0F0xBC_xD820_0CA4[CUR_TEMP]. Separate controls are provided for increasing and decreasing temperatures. The latest measured temperature is referred to as T_{ctlm} below.

If downward slew control is enabled (D0F0xBC_xD820_0CA4[TMP_SLEW_DN_EN]), T_{ctl} is not updated down unless T_{ctlm} remains below T_{ctl} for a time specified by D0F0xBC_xD820_0CA4[PER_STEP_TIME_DN]. If at any point before the timer expires T_{ctlm} equals or exceeds T_{ctl}, then the timer resets and T_{ctl} is not updated. If the timer expires, then T_{ctl} is reduced by 0.125. If downard slew control is disabled, then if T_{ctlm} is less than T_{ctl}, T_{ctl} is immediately updated to T_{ctlm}.

The upward slew control works similar to downward slew control except that if T_{ctlm} exceeds T_{ctl} by a value defined by D0F0xBC_xD820_0CA4[TMP_MAX_DIFF_UP] then T_{ctl} is immediately updated to T_{ctlm}. Otherwise, T_{ctlm} must remain above T_{ctl} for time specified by D0F0xBC_xD820_0CA4[PER_STEP_TIME_UP] before T_{ctl} is incremented by 0.125.

2.10.3 Temperature-Driven Logic

The temperature calculated by the TCC is used by HTC,THERMTRIP, and PROCHOT_L.

2.10.3.1 PROCHOT_L and Hardware Thermal Control (HTC)

The processor *HTC-active state* is characterized by (1) the assertion of PROCHOT_L, (2) reduced power con-

sumption, and (3) reduced performance. While in the HTC-active state, software should not change the following: All D0F0xBC_xD820_0C64 fields (except for HTC_ACTIVE and HTC_EN), D0F0xBC_xD820_0C64[DIS_PROCHOT_PIN]. Any change to the previous list of fields when in the HTC-active state can result in undefined behavior. HTC status and control is provided through D0F0xBC_xD820_0C64.

The PROCHOT_L pin acts as both an input and as an open-drain output. As an output, PROCHOT_L is driven low to indicate that the HTC-active state has been entered due to an internal condition, as described by the following text. The minimum assertion and deassertion time for PROCHOT_L is 200 us with a minimum period of 2 ms. APU power is lowered for a period up to 1ms following the assertion of PROCHOT_L.

While in the HTC-active state, the following power reduction actions are taken:

- CPU cores are limited to a P-state (specified by D0F0xBC_xD820_0C64[HTC_PSTATE_LIMIT]); see 2.5.2 [CPU Core Power Management].
- The NB is placed in the lowest performance P-states supported by the processor.
- The GPU graphics unit and multi-media units may be placed in the low power DPM0 state based on the state of D0F0xBC_xD820_0C64[HtcToGnbEn] and D0F0xBC_xD820_0C64[ProcHotToGnbEn].

The processor enters the HTC-active state if all of the following conditions are true:

- D0F0xBC_xD820_0C64[HTC_EN] == 1.
- PWROK == 1.
- D0F0xBC_xD820_0CE4[THERM_TP] == 1.

and any of the following conditions are true:

- T_{ctl} is greater than or equal to the HTC temperature limit (D0F0xBC_xD820_0C64[HTC_TMP_LMT]).
- PROCHOT_L == 0.

The processor exits the HTC-active state when all of the following are true:

- T_{ctl} is less than the HTC temperature limit (D0F0xBC_xD820_0C64[HTC_TMP_LMT]).
- T_{ctl} has become less than the HTC temperature limit (D0F0xBC_xD820_0C64[HTC_TMP_LMT]) minus the HTC hysteresis limit (D0F0xBC_xD820_0C64[HTC_HYST_LMT]) since being greater than or equal to the HTC temperature limit (D0F0xBC_xD820_0C64[HTC_TMP_LMT]).
- PROCHOT_L == 1.

The default value of the HTC temperature threshold (T_{ctl_max}) is specified in the Power and Thermal Datasheet.

2.10.3.2 Software P-state Limit Control

D18F3x68 [Software P-state Limit] provides a software mechanism to limit the P-state MSRC001_0061[CurPstateLimit]. See 2.5.2 [CPU Core Power Management].

2.10.3.3 THERMTRIP

If the processor supports the THERMTRIP state (as specified by D0F0xBC_xD820_0CE4[THERM_TP_EN]) and the temperature approaches the point at which the processor may be damaged, the processor enters the THERMTRIP state. The THERMTRIP function is enabled after cold reset (after PWROK asserts and RESET_L deasserts). It remains enabled in all other processor states, except during warm reset (while RESET_L is asserted). The THERMTRIP state is characterized as follows:

- Nearly all clocks are gated off to reduce dynamic power.
- A low-value VID is generated.

- The system is placed into the S5 ACPI state (power off).

A cold reset is required to exit the THERMTRIP state.

2.11 Root Complex

2.11.1 Overview

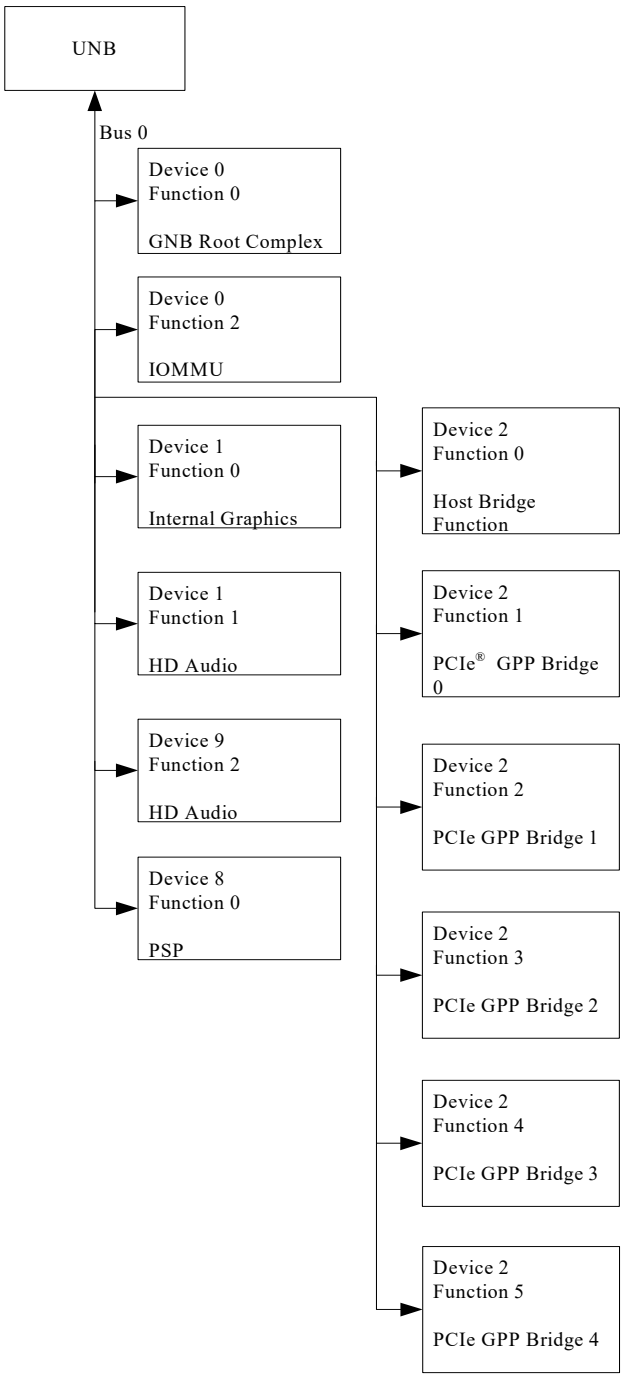


Figure 6: Root Complex Topology

2.11.2 Interrupt Routing

The GNB includes a fully programmable IOAPIC. The IOAPIC registers are accessed through the D0F0xF8 index and D0F0xFC data pair registers using two back-to-back config cycles. PCI defined INTx interrupts for

each bridge are routed to IOAPIC pins via the bridge interrupt routing registers located at D0F0xFC_x1[4:0].

2.11.2.1 IOAPIC Configuration

The IOAPIC configuration is performed by the following sequence:

1. Set the base address for the memory mapped registers by programming D0F0xFC_x01[IoapicAddr] and D0F0xFC_x02[IoapicAddrUpper].
2. Enable IOAPIC by programming D0F0xFC_x00[IoapicEnable] = 1.
3. Only if the system is in PIC mode, program D0F0xFC_x00[IoapicSbFeatureEn] = 1. This bit should be programmed to 0 when the system is in APIC mode.

The IOAPIC has a total of 62 interrupt inputs. These inputs are as follows:

- There are 10 groups of PCIe® interrupts (each group having a 4-bit external interrupt bus (INT A/B/C/D) and a 1-bit bridge interrupt).
- a 4-bit external interrupt bus from SPG, ACG and gBIF.

A recommended interrupt routing and swizzling configuration is shown below.
The recommended interrupt routing and swizzling configuration is as shown in Table 53.

Table 53: Recommended Interrupt Routing and Swizzling Configuration

Device	Register	Setting	Description
Dev2Fn1	D0F0xFC_x10[BrExtIntrGrp]	0h	Map INT A/B/C/D to interrupt 0/1/2/3. Map bridge interrupt to interrupt 23.
	D0F0xFC_x10[BrExtIntrSwz]	0h	
	D0F0xFC_x10[BrIntIntrMap]	17h	
Dev2Fn2	D0F0xFC_x11[BrExtIntrGrp]	2h	Map INT A/B/C/D to interrupt 8/9/10/11. Map bridge interrupt to interrupt 24.
	D0F0xFC_x11[BrExtIntrSwz]	0h	
	D0F0xFC_x11[BrIntIntrMap]	18h	
Dev2Fn3	D0F0xFC_x12[BrExtIntrGrp]	4h	Map INT A/B/C/D to interrupt 16/17/18/19. Map bridge interrupt to interrupt 25.
	D0F0xFC_x12[BrExtIntrSwz]	0h	
	D0F0xFC_x12[BrIntIntrMap]	19h	
Dev2Fn4	D0F0xFC_x13[BrExtIntrGrp]	6h	Map INT A/B/C/D to interrupt 24/25/26/27. Map bridge interrupt to interrupt 26.
	D0F0xFC_x13[BrExtIntrSwz]	0h	
	D0F0xFC_x13[BrIntIntrMap]	1Ah	
Dev2Fn5	D0F0xFC_x14[BrExtIntrGrp]	0h	Map INT A/B/C/D to interrupt 3/0/1/2. Map bridge interrupt to interrupt 23.
	D0F0xFC_x14[BrExtIntrSwz]	1h	
	D0F0xFC_x14[BrIntIntrMap]	17h	
Dev8Fn0	D0F0xFC_x30[SpgExtIntrGrp]	3h	Map INT A/B/C/D to interrupt 14/15/12/13.
	D0F0xFC_x30[SpgExtIntrSwz]	2h	
Dev9Fn2	D0F0xFC_x2F[AcgPortAExtIntGrp]	5h	Map INT A/B/C/D to interrupt 22/23/20/21.
	D0F0xFC_x2F[AcgPortAExtIntrSwz]	2h	
GBIF	D0F0xFC_x0F[GBIFExtIntrGrp]	1h	Map INT A/B/C/D to interrupt 6/7/4/5.
	D0F0xFC_x0F[GBIFExtIntrSwz]	2h	

2.11.3 Links

2.11.3.1 Overview

There is one 5-port 8-lane (5x8) PCIe® core for General Purpose Ports (GPP).
Each multi-lane port can be limited to lower link widths for applications that require fewer lanes. All PCIe links are capable of supporting 2.5/5.0 GT/s data rates.

Each port has a Type 1 Virtual PCI-to-PCI bridge header in the PCI configuration space mapped to devices according to Figure 6.

2.11.3.2 Link Configurations

The following link configurations are supported for the GPP links:

Table 54: Supported General Purpose (GPP) Link Configurations

D0F0xE4	GPP Port Lane							
x0150_0080	7	6	5	4	3	2	1	0
0000_0001h	x4 Link				x4 Link			
0000_0002h	x2 Link		x2 Link		x4 Link			
0000_0003h	x1 Link	x1 Link	x2 Link		x4 Link			
0000_0004h	x1 Link	x1 Link	x1 Link	x1 Link	x4 Link			
0000_0004h	—	—	x1 Link	x1 Link	x1 Link	x1 Link	x2 Link	
0000_0004h	—	—	x1 Link	x1 Link	x2 Link		x1 Link	x1 Link
0000_0004h	—	—	x2 Link		x1 Link	x1 Link	x1 Link	x1 Link

2.11.4 Root Complex Configuration

2.11.4.1 LPC MMIO Requirements

To ensure proper operation of LPC generated DMA requests, the UMI must be configured to send processor generated MMIO writes that target the LPC bus to the FCH as non-posted writes. To ensure this requirement the MMIO address space of the LPC bus must not be included in the ranges specified by D18F1x[2CC:2A0,1CC:180,BC:80] [MMIO Base/Limit] and non-posted protocol for memory writes must be enabled using the following sequence before LPC DMA transactions are initiated.

1. Configure the FCH to use the non-posted write protocol. See the FCH register specification for configuration details.
2. Locate the PCIe® core that has the UMI link (read D0F0x64_x1F to get location of the FCH).
3. Note that this step should only be performed for the PCIe core with the UMI link (found via step 2 above). Program D0F0xE4_x0140_0010[UmiNpMemWrite] = 1.
4. Program D0F0x98_x06[UmiNpMemWrEn] = 1.

The FP4 package includes an integrated FCH and does not have a UMI link. Hence, the below LPC requirements apply to this package.
To ensure proper operation of LPC generated DMA requests, the FCH must be configured to send processor

generated MMIO writes that target the LPC bus to the FCH as non-posted writes. The MMIO address space of the LPC bus must not be included in the ranges specified by D18F1x[2CC:2A0,1CC:180,BC:80] [MMIO Base/Limit] and non-posted protocol for memory writes must be enabled using the following programming before LPC DMA transactions are initiated.

- Program D0F0x98_x06[UmiNpMemWrEn] = 1.

2.11.4.2 Configuration for non-FCH Bridges

BIOS should program the following in non-FCH bridges:
1. Program D0F0xCC_x01_ib[23:19,15:11][CrsEnable] = 1 for D0F0xC8[NbDevIndSel] == 11h-15h, 19h-1Ah.
2. Program D0F0xCC_x01_ib[23:19,15:11][SetPowEn] = 1 for D0F0xC8[NbDevIndSel] == 11h-15h, 19h-1Ah.

2.11.4.3 Link Configuration and Initialization

Link configuration and initialization is performed by the following sequence:
1. 2.11.4.3.1 [Link Configuration and Core Initialization]
2. 2.11.4.3.2 [Link Training]
3. 2.11.4.5 [Link Power Management]
4. Lock link configuration registers.
 - Program D0F0xE4_x0140_0010[HwInitWrLock] = 1.
 - Program D0F0x64_x00[HwInitWrLock] = 1.
5. IF(an external FCH connected through UMI using D0F0xC8[NbDevIndSel]) THEN program D0F0xCC_x01_ib[23:19,15:11][CfgDis] = 1.

2.11.4.3.1 Link Configuration and Core Initialization

Link configuration is done on a per link basis. Lane reversal, IO link selection, and lane enablement is configured through this sequence.

2.11.4.3.2 Link Training

Link training is performed on a per link basis. BIOS may train the links in parallel.

2.11.4.4 Miscellaneous Features

2.11.4.4.1 Lane Reversal

Normally, the lanes of each port are physically numbered from n-1 to 0 where n is the number of lanes assigned to the port. Physical lane numbering can be reversed according to the following methods:
- To reverse the physical lane numbering for a specific port, program D2F[5:1]xE4_xC1[StrapReverseLanes] = 1.
- To reverse the physical lane numbering for all ports in the GPP interfaces, program D0F0xE4_x0140_00C0[StrapReverseAll] = 1.

Note that logical port numbering is established during link training regardless of the physical lane numbering.

2.11.4.4.2 Link Speed Changes

Link speed changes can only occur on Gen2 and Gen2/Gen3 capable links. To verify that Gen2/Gen3 speeds

are supported verify D2F[5:1]x64[LinkSpeed] == 02h or D2F[5:1]x64[LinkSpeed] == 03h. Note that Gen3 support is only for the graphics link.

2.11.4.4.2.1 Autonomous Link Speed Changes

To enable autonomous speed changes on a per port basis:
1. Program D2F[5:1]x88[TargetLinkSpeed] = 2h.
2. Program D0F0xE4_x0150_0[C:8]03[StrapBifDeemphasisSel] = 1.
3. Program D2F[5:1]xE4_xA4[LcGen2EnStrap] = 1.
4. Program D2F[5:1]xE4_xC0[StrapAutoRcSpeedNegotiationDis] = 0.
5. Program D2F[5:1]xE4_xA4[LcMultUpstreamAutoSpdChngEn] = 1.
6. Program D2F[5:1]xE4_xA2[LcUpconfigureDis] = 0.

To enable autonomous speed changes on a per port basis for Gen3:
1. Program D2F[5:1]x88[TargetLinkSpeed] = 3h.
2. Program D2F[5:1]xE4_xA4[LcGen3EnStrap] = 1.
3. Program D2F[5:1]xE4_xC0[StrapAutoRcSpeedNegotiationDis] = 0.

2.11.4.4.3 Deemphasis

Deemphasis strength can be changed on a per-port basis by programming D2F[5:1]xE4_xB5[LcSelectDeemphasis].

2.11.4.5 Link Power Management

2.11.4.5.1 Link States

To enable support for L1 program D2F[5:1]xE4_xA0[LcL1Inactivity] = 6h.

To enable support for L0s:
- Program D2F[5:1]xE4_xA1[LcDontGotoL0sifL1Armed] = 1.
- Program D2F[5:1]xE4_xA0[LcL0sInactivity] = 9h.

2.11.4.5.2 Dynamic Link-width Control

Dynamic link-width control is a power saving feature that reconfigures the link to run with fewer lanes. The inactive lanes are turned off to conserve power.

Each link can switch among widths of: x1, x2, x4, x8, and x16, up to the maximum port width.

The link width is controlled by the following mechanism:
- Up/Down Reconfiguration: The link is retrained according to the PCI Express® specification.

The core has the capability to turn off the inactive lanes of trained links. To enable this feature program D2F[5:1]xE4_xA2[LcDynLanesPwrState] = 11b.

2.11.4.6 Link Test and Debug Features

2.11.4.6.1 Compliance Mode

To enable Gen1 software compliance mode program D2F[5:1]xE4_xC0[StrapForceCompliance] = 1 for each

port to be placed in compliance mode.

To enable Gen2 software compliance mode:
1. BIOS enables Gen2 capability by programming D0F0xE4_x0140_00C1[StrapGen2Compliance] = 1.
2. Program D2F[5:1]xE4_xA4[LcGen2EnStrap] = 1.
3. Program D2F[5:1]x88[TargetLinkSpeed] = 2h for each port to be placed in compliance mode.
4. Program D2F[5:1]x88[EnterCompliance] = 1 for each port to be placed in compliance mode.

To enable Gen3 software compliance mode:
1. BIOS enables Gen3 capability on the link by programming D0F0xE4_x0140_00C1[StrapGen3Compliance] = 1.
2. Program D2F[5:1]xE4_xA4[LcGen3EnStrap] = 1.
3. Program D2F[5:1]x88[TargetLinkSpeed] = 3h for each port to be placed in compliance mode.
4. Program D2F[5:1]x88[EnterCompliance] = 1 for each port to be placed in compliance mode.

2.11.5 FCH Messages

To replace the wires and messages previously used between the processor and the FCH, upstream and downstream messages are defined through a combination of messages and reads or posted writes of special addresses. These message packets look like regular PCIe® packets, but are AMD proprietary packets across the UMI link.

2.11.6 BIOS Timer

The root complex implements a 32-bit microsecond timer (see D0F0xE4_x0130_80F0 and D0F0xE4_x0130_80F1) that the BIOS can use to accurately time wait operations between initialization steps.

To ensure that BIOS waits a minimum number of microseconds between steps BIOS should always wait for one microsecond more than the required minimum wait time.

2.11.7 PCIe® Client Interface Control

This interface is accessed through the indexed space registers located at D0F2xF8 within the Device 0 Function 2 (IOMMU) Configuration Registers.

BIOS should perform the following steps to initialize the interface:

1. Program D0F0x64_x0D[PciDev0Fn2RegEn] = 1h.
2. Program credits for the PPx40 client as follows:
 A. Program D0F2xFC_x32_L1i[0][DmaNpHaltDis] = 0h.
 B. Program D0F2xFC_x32_L1i[0][DmaBufCredits] = 20h.
 C. Program D0F2xFC_x32_L1i[0][DmaBufMaxNpCred] = 1Fh.
3. Program credits for the BIF client as follows:
 A. Program D0F2xFC_x32_L1i[2][DmaNpHaltDis] = 1h.
 B. Program D0F2xFC_x32_L1i[2][DmaBufCredits] = 20h.
 C. Program D0F2xFC_x32_L1i[2][DmaBufMaxNpCred] = 20h.
4. Program credits for the IOAGR client as follows:
 A. Program D0F2xFC_x32_L1i[3][DmaNpHaltDis] = 1h.
 B. Program D0F2xFC_x32_L1i[3][DmaBufCredits] = 20h.
 C. Program D0F2xFC_x32_L1i[3][DmaBufMaxNpCred] = 20h.
5. Program clock gating as follows:

A. Program D0F2xFC_x33_L1i[3:0][L1DmaClkgateEn] = 1h.
B. Program D0F2xFC_x33_L1i[3:0][L1CacheClkgateEn] = 1h.
C. Program D0F2xFC_x33_L1i[3:0][L1CpslvClkgateEn] = 1h.
D. Program D0F2xFC_x33_L1i[3:0][L1DmaInputClkgateEn] = 1h.
E. Program D0F2xFC_x33_L1i[3:0][L1PerfClkgateEn] = 1h.
F. Program D0F2xFC_x33_L1i[3:0][L1MemoryClkgateEn] = 1h.
G. Program D0F2xFC_x33_L1i[3:0][L1RegClkgateEn] = 1h.
H. Program D0F2xFC_x33_L1i[3:0][L1HostreqClkgateEn] = 1h.
I. Program D0F2xFC_x33_L1i[3:0][L1L2ClkgateEn] = 1h.
J. For L1 IOAGR only, program D0F2xFC_x3D_L1i[3:0][14] = 1h.
K. Program D0F2xF4_x33[CKGateL2ARegsDisable] = 0h.
L. Program D0F2xF4_x33[CKGateL2ADynamicDisable] = 0h.
M. Program D0F2xF4_x33[CKGateL2ACacheDisable] = 0h.
N. Program D0F2xF4_x90[CKGateL2BRegsDisable] = 0h.
O. Program D0F2xF4_x90[CKGateL2BDynamicDisable] = 0h.
P. Program D0F2xF4_x90[CKGateL2BMiscDisable] = 1h.

6. Program D0F0x64_x0D[PciDev0Fn2RegEn] = 0h.

2.12 IOMMU

The processor includes an IOMMU revision 2. See the *AMD I/O Virtualization Technology™ (IOMMU) Specification*, order #48882, referneced in 1.2 [Reference Documents].

2.12.1 IOMMU Configuration Space

The IOMMU configuration space consists of the following four groups:

- PCI Configuration space. See 3.4 [Device 0 Function 2 (IOMMU) Configuration Registers].
- IOMMU Memory Mapped Register space. See 3.17 [IOMMU Memory Mapped Registers].
- IOMMU L1 Indexed space accessed through D0F2xF8 [IOMMU L1 Config Index].
- IOMMU L2 Indexed space accessed through D0F2xF0 [IOMMU L2 Config Index].

2.12.2 IOMMU Initialization

BIOS should perform the following steps to initialize the IOMMU:

1. Program D0F0x64_x0D[PciDev0Fn2RegEn] = 1h.
2. Program D0F2x44 [IOMMU Base Address Low] and D0F2x48 [IOMMU Base Address High] to allocate a 512K region of MMIO space for IOMMU memory mapped registers. This region of MMIO space is reserved for IOMMU and BIOS must not allocate it for use by system software.
3. Program D0F2x50[IommuHtAtsResv] = 0h.
4. Program D0F2x44[IommuEnable] = 1h.
5. Program D0F2x7C[MsiMultMessCapW] = 2h.
6. Program D2F[5:1]xE4_xC1[StrapExtendedFmtSupported] = 1 and D2F[5:1]xE4_xC1[StrapE2EPrefixEn] = 1.
7. Program IOMMUx18[Isoc] = 1h if processors support isochronous channel.
8. Program the registers with BIOS recommendations in L1 (D0F2xFC) and L2 (D0F2xF4) indexed space. See 2.11.7 [PCIe® Client Interface Control].
9. Check if any PCIe® devices in the system support the Phantom Function. For each PCIe core that none of its connected device advertising support for the Phantom Function, program D0F2xFC_x07_L1i[3:0][0] = 1 for the L1 corresponding to that PCIe core.
10. If a PCIe port is hot-plug capable, then program D0F2xFC_x07_L1i[3:0][0] = 0 for the L1 corresponding to the PCIe core.
11. If at least one PCIe to PCI-X® bridge exists on a PCIe port or a HotPlug capable PCIe slot is present on a PCIe port then program D0F2xFC_x0D_L1i[3:0][VOQPortBits] = 111b for the L1 corresponding to the particular PCIe core.
12. Program the location of the SBinto D0F2xF4_x49 [L2_SB_LOCATION]. The program value is required to match the value programmed in D0F0x64_x1F [FCH Location].
13. Program the location of the SB into D0F2xFC_x09_L1i[3:0] [L1_SB_LOCATION] for the L1 corresponding to the iFCH or the PCIe core which FCH is located. The program value is required to match the value programmed in D0F0x64_x1F [FCH Location]. Leave register at default value for all L1s corresponding to other PCIe cores.

2.12.2.1 IOMMU L1 Initialization

BIOS should perform the following steps to initialize the IOMMU L1:

1. Program D0F2xFC_x0C_L1i[3:0][L1VirtOrderQueues] = 4h.
2. Program D0F2xFC_x32_L1i[3:0][AtsMultipleRespEn] = 1h.

3. Program D0F2xFC_x32_L1i[3:0][AtsMultipleL1toL2En] = 1h.
4. Program D0F2xFC_x32_L1i[3:0][TimeoutPulseExtEn] = 1h.
5. Program D0F2xFC_x07_L1i[3:0][AtsPhysPageOverlapDis] = 1h.
6. Program D0F2xFC_x07_L1i[3:0][AtsSeqNumEn] = 1h.
7. Program D0F2xFC_x07_L1i[3:0][SpecReqFilterEn] = 1h.
8. Program D0F2xFC_x07_L1i[3:0][L1NwEn] = 1h.
9. Program D0F2xFC_x07_L1i[3:0][CacheFilterEn] = 1h.

2.12.2.2 IOMMU L2 Initialization

BIOS should perform the following steps to initialize the IOMMU L2:

1. Program D0F2xF4_x10[DTCInvalidationSel] = 2h.
2. Program D0F2xF4_x14[ITCInvalidationSel] = 2h.
3. Program D0F2xF4_x18[PTCAInvalidationSel] = 2h.
4. Program D0F2xF4_x50[PDCInvalidationSel] = 2h.
5. Program D0F2xF4_x80[ERRRuleLock0] = 1h.
6. Program D0F2xF4_x30[ERRRuleLock1] = 1h.
7. Program D0F2xF4_x34[L2aregHostPgsize] = 2h.
8. Program D0F2xF4_x34[L2aregGstPgsize] = 2h.
9. Program D0F2xF4_x94[L2bregHostPgsize] = 2h.
10. Program D0F2xF4_x94[L2bregGstPgsize] = 2h.
11. Program D0F2xF4_x4C[GstPartialPtcCntrl] = 3h.
12. If all ports on a particular L1 are disabled, program the corresponding bit of D0F2xF4_x57 = 1h.
13. Program D0F2xF4_x57[1] = 1h.
14. Program D0F2xF4_x56[CPFlushOnWait] = 1h.
15. Program D0F2xF4_x56[CPFlushOnInv] = 0h.
16. Program D0F2xF4_x53[L2bUpdateFilterBypass] = 0h.
17. Program D0F2xF4_x22[L2aUpdateFilterBypass] = 0h.
18. Program D0F2xF4_x54[TWPrefetchRange] = 7h.
19. Program D0F2xF4_x54[TWPrefetchEn] = 1h.

2.12.2.3 IOMMU SMI Filtering

In order to ensure system management interrupts come from valid peripheral sources, the IOMMU supports an SMI filter (IOMMUx30[SmiFSup] == 01b). SMI interrupts are filtered according to the values programmed in the SMI filter registers. The registers specify what sources are allowed to send SMI interrupts. IOMMUx30[SmiFRC] indicates the number of SMI filter registers available.

The BIOS should set up the SMI filter registers and lock them. These settings take effect when system software enables the IOMMU and enables SMI filtering. BIOS should perform the following steps to set up SMI filtering:

1. Choose an SMI filter register from the available set described in IOMMUx30[SmiFRC]. Select one register for each SMI source.
2. Program selected SMI filter register to the Device ID of the peripheral issuing the SMI interrupts via IOMMUx[78,70,68,60][SMIDid]. Program one register for each SMI source.
3. Program selected SMI filter register to be valid via IOMMUx[78,70,68,60][SMIDV]. Program one register for each SMI source.
4. Program selected SMI filter register to be locked via IOMMUx[78,70,68,60][SMIFlock]. Program one register for each SMI source.

2.12.2.4 IOMMU Power Gating

2.12.2.4.1 Static Power Down IOMMU L2

When IOMMU is in pass through mode, IOMMU L2 can be power down to conserve power using PGFSM. Once IOMMU L2 is powered down, a reset is required in order to power IOMMU L2 back on.

1. Program D0F2xF4_x3B[FsmAddr] = FFh.
2. Program D0F2xF4_x3B[PowerDown] = 1h.
3. Program D0F2xF4_x3B[PowerDown] = 0h to generate a pulse.

2.12.2.4.2 Memory Power Gating in IOMMU L1

The IOMMU L1 memories support dynamic power gating to conserve power. Three dynamic power gating modes are supported: Light Sleep (LS), Deep Sleep (DS) or shut down (SD):

1. Program D0F2xFC_x3A_L1i[3:0][LSThres] = C8h.
2. Program D0F2xFC_x3B_L1i[3:0][DSThres] = C8h.
3. Program D0F2xFC_x3C_L1i[3:0][SDThres] = 44Ch.
4. Program D0F2xFC_x39_L1i[3:0][LSEn] = 1h.
5. Program D0F2xFC_x39_L1i[3:0][DSEn] = 1h.
6. Program D0F2xFC_x39_L1i[3:0][SDEn] = 1h.

2.13 System Management Unit (SMU)

The system management unit (SMU) is a subcomponent of the northbridge that is responsible for a variety of system and power management tasks during boot and runtime. The SMU uses two blocks, System Management Controller (SMC) and Platform Security Processor (PSP), in order to assist with many of these tasks. At the architectural level, PSP is known as MP0 and SMC is known as MP1.

2.13.1 System Management Controller (SMC)

The SMC is a standalone complex within AMD Family 15h Models 70h-7Fh processors that is responsible for maintaining the power management environment. Its functions include dynamic power management, state switching and fan control. The SMC contains a microcontroller to assist with many of these tasks.

2.14 Graphics Processor (GPU)

The APU contains an integrated DirectX® 11/12 compliant graphics processor.

2.14.1 Graphics Memory Controller (GMC)

The graphics memory controller is responsible for servicing memory requests from the different blocks within the GPU and forwarding routing them to the appropriate interface. The GMC is also responsible for translating GPU virtual address to GPU physical addresses and for translating GPU physical addresses to system addresses.

2.14.2 Frame Buffer (FB)

2.14.2.1 General Guidelines

The frame buffer is defined as the portion of system memory dedicated for reading and writing of display information.

Table 55: Recommended Legacy Frame Buffer Configurations

System Memory Size	Frame Buffer Size
1 x 2 GB	UMA = 256 MB
1 x 4 GB	UMA = 512 MB
2 GB + 4 GB = 6 GB	UMA = 1 GB
2 x 4 GB = 8 GB	UMA = 1 GB

Table 56: Non-Legacy Mode (Default for Windows® 8.1 and Windows® 10) Carve Out Table

System Memory Size	Resolution		
	1920x1080 and below	2560x1600	3840x2160
2 GB and 3 GB	32 MB	48 MB	80 MB
4 GB	80 MB	80 MB	80 MB
6 GB	384 MB	384 MB	384 MB
8 GB	512 MB	512 MB	512 MB

2.14.2.2 Host Data Path Guidelines

The HDP block provides host access to the frame buffer address translation.

- HDP size should be rounded up to next power of 2 (up to 256 MB) controlled by:
 - D0F0xD4_x0109_14E1[StrapBifMemApSize] and D0F0xD4_x0109_1507[StrapBifMemApSizePin].
- UMA size = ({GMMx2024[FbTop], FFFFFFh} + 1) - {GMMx2024[FbBase], 000000h}.

Table 57: Frame Buffer Configuration Examples

System Memory Size	Resolution		
	1920x1080 and below	2560x1600	3840x2160
>= 2 GB & <4 GB	UMA = 32 MB	UMA = 48 MB	UMA = 80 MB
	HDP = 64 MB	HDP = 64 MB	HDP = 128 MB
	GMMx224[C:4][L1_TLB_DEBUG[0]] = 1		
>=4 GB	UMA = 96 MB	UMA = 112 MB	UMA = 144 MB
	HDP = 128 MB	HDP = 128 MB	HDP = 256 MB
	GMMx224[C:4][L1_TLB_DEBUG[0]] = 1		

2.15 Audio Processor (ACG)

2.15.1 Programming ACG Host Address Mapping

The audio controller exposes standard PCI BARs to request/claim MMIO resources. The standard memory access enable register must also be set in the configuration header to enable MMIO access. In order to forward host MMIO requests in the system physical address space on to the ARM® AMBA AXI bus, the address mapping must be configured and enabled.

1. Program D9F2xF4_x10[BarMap] = 0001_0000h.
2. Program D9F2xF4_x11[BarMap] = 0h.
3. Program D9F2xF4_x12[BarMap] = 0001_4000h.

4. Program D9F2xF4_x13[BarMap] = 0h.
5. Program D9F2xF4_x0F[Bar0MapLock] = 1b.
6. Program D9F2xF4_0xF[Bar1MapLock] = 1b.

2.15.2 Programming ACG DMA Apertures

The DMA aperture must be configured to enable DMA access from the AXI interface.

1. Program D9F2xF4_x28[31:24] = 00h.
2. Program D9F2xF4_x29[Aper0Base] = 00h.
3. Program D9F2xF4_x2A[31:24] = FFh.
4. Program D9F2xF4_x2B[Aper0Limit] = FFFFh.
5. Program D9F2xF4_x2C[31:24] = 00h.
6. Program D9F2xF4_x2D[Aper0Map] = 00h.
7. Enable PASID to support nested paging in IOMMU by programming D9F2xF4_x21[PasidEn] = 1b and D9F2xF4_x21[Pasid] = Desired PASID.
8. Program D9F2xF4_x20[Aper0En] = 1b.
9. Program D9F2xF4_x20[Aper0Lock] = 1b.
10. Program D9F2x04[BusMasterEn] = 1b.

To expose the PCI advanced capabilities register, program D9F2x44[PmNxtPtrW] = A4h.
The Audio Controller (D9F2) can be disabled through program D0F0x64_x50[7:4] = Fh. It terminates the HOST requests and DMA requests to/from Audio Controller. Before disable the function,system can enable the ACG and Audio controller clock gating to save the power after the function is disabled.
1. Program D9F2xEC_x4A[31:24] = 00h.
2. Program D9F2xEC_x4C[31:24] = 00h.

For the Audio Controller itself, please refer to the Audio Controller clock gating registers and they can be accessed through the BAR (D9F2x14).

To enable sampling of the interrupt line from the AXI interface, program D9F2xF4_x49[0] = 1b.

2.16 RAS Features

2.16.1 Machine Check Architecture

The processor contains logic and registers to detect, log, and correct errors in the data or control paths in each core and the Northbridge. The Machine Check Architecture (MCA) defines the facilities by which processor and system hardware errors are logged and reported to system software. This allows system software to perform a strategic role in recovery from and diagnosis of hardware errors.

Refer to the AMD64 Architecture Programmer's Manual for an architectural overview and methods for determining the processor's level of MCA support. See 1.2 [Reference Documents].

The ability of hardware to generate a machine check exception upon an error is indicated by CPUID Fn0000_0001_EDX[MCE] or CPUID Fn8000_0001_EDX[MCE].

2.16.1.1 Machine Check Registers

CPUID Fn0000_0001_EDX[MCA] or CPUID Fn8000_0001_EDX[MCA] indicates the presence of the following machine check registers:

- MSR0000_0179 [Global Machine Check Capabilities (MCG_CAP)]
 - Reports how many machine check register banks are supported.
- MSR0000_017A [Global Machine Check Status (MCG_STAT)]
 - Provides basic information about processor state after the occurence of a machine check error.
- MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)]
 - Used by software to enable or disable the logging and reporting of machine check errors in the error-reporting banks.

The error-reporting machine check register banks supported in this processor are:

- MC0: Load-store unit (LS), including data cache.
- MC1: Instruction fetch unit (IF), including instruction cache.
- MC2: Combined unit (CU), including L2 cache.
- MC3: Reserved.
- MC4: Northbridge (NB), including the IO link. There is only one NB error reporting bank, independent of the number of cores.
- MC5: execution unit (EX), including mapper/scheduler/retire/execute functions and fixed-issue reorder buffer.
- MC6: Floating point unit (FP).

- The register types within each bank are:
 - MCi_CTL, Machine Check Control: Enables error reporting via machine check exception. The MC*i*_CTL register in each bank must be enabled by the corresponding enable bit in MCG_CTL (MSR0000_017B).
 - MCi_STATUS, Machine Check Status: Logs information associated with errors.
 - MCi_ADDR, Machine Check Address: Logs address information associated with errors.
 - MCi_MISC, Machine Check Miscellaneous: Log miscellaneous information associated with errors, as defined by each error type.
 - MCi_CTL_MASK, Machine Check Control Mask: Inhibit detection of an error source unless otherwise specified.

The following table identifies the registers associated with each error-reporting machine check register bank:

Table 58: MCA Register Cross-Reference Table

Register Bank (MC*i*)	MCA Register				
	CTL	STATUS	ADDR	MISC	CTL_MASK
MC0	MSR0000_0400	MSR0000_0401	MSR0000_0402	MSR0000_0403	MSRC001_0044
MC1	MSR0000_0404	MSR0000_0405	MSR0000_0406	MSR0000_0407	MSRC001_0045
MC2	MSR0000_0408	MSR0000_0409	MSR0000_040A	MSR0000_040B	MSRC001_0046
MC3	MSR0000_040C	MSR0000_040D	MSR0000_040E	MSR0000_040F	MSRC001_0047
MC4	MSR0000_0410	MSR0000_0411	MSR0000_0412	MSR0000_0413	MSRC001_0048
MC5	MSR0000_0414	MSR0000_0415	MSR0000_0416	MSR0000_0417	MSRC001_0049
MC6	MSR0000_0418	MSR0000_0419	MSR0000_041A	MSR0000_041B	MSRC001_004A

Corrected, deferred, and uncorrected errors are logged in MCi_STATUS and MCi_ADDR as they occur. Uncorrected errors that are enabled in MCi_CTL result in a Machine Check exception.

Each MCi_CTL register must be enabled by the corresponding enable bit in MSR0000_017B [Global Machine Check Exception Reporting Control (MCG_CTL)].

MCi_CTL_MASK allow BIOS to mask the presence of any error source from software for test and debug. When error sources are masked, it is as if the error was not detected. Such masking consequently prevents error responses and actions.

Each MCA bank implements a number of machine check miscellaneous registers, denoted as MCi_MISCj, where j goes from 0 to a maximum of 8. If there is more than one MCi_MISC register in a given bank, a non-zero value in MCi_MISC0[BlkPtr] points to the contiguous block of additional registers.

The presence of valid information in the first MISC register in the bank (MC*i*_MISC0) is indicated by MC*i*_STATUS[MiscV]. The presence of valid information in additional implemented MISC registers is indicated by MC*i*_MISC*j*[Val] in the target register.

2.16.1.2 Machine Check Errors

The classes of machine check errors are, in priority order from highest to lowest:

- Uncorrected
- Deferred
- Corrected

Uncorrected errors cannot be corrected by hardware and may cause loss of data, corruption of processor state, or both. Uncorrected errors update the status and address registers if not masked from logging in MCi_CTL_MASK. Information in the status and address registers from a previously logged lower priority error is overwritten. Previously logged errors of the same priority are not overwritten. Uncorrected errors that are enabled in MCi_CTL result in reporting to software via machine check exceptions. If an uncorrected error is masked from logging, the error is ignored by hardware (exceptions are noted in the register definitions). If an uncorrected error is disabled from reporting, containment of the error and logging/reporting of subsequent errors may be affected. Therefore, enable reporting of unmasked uncorrected errors for normal operation. Disable reporting of uncorrected errors only for debug purposes.

Deferred errors are errors that cannot be corrected by hardware, but do not cause an immediate interruption in program flow, loss of data integrity, or corruption of processor state. These errors indicate that data has been corrupted but not consumed; no exception is generated because the data has not been referenced by a core or an IO link. Hardware writes information to the status and address registers in the corresponding bank that identifies the source of the error if deferred errors are enabled for logging. If there is information in the status and address registers from a previously logged lower priority error, it is overwritten. Previously logged errors of the same or higher priority are not overwritten. Deferred errors are not reported via machine check exceptions; they can be seen by polling the MCi_STATUS registers.

Corrected errors are those which have been corrected by hardware and cause no loss of data or corruption of processor state. Hardware writes the status and address registers in the corresponding register bank with information that identifies the source of the error if they are enabled for logging. Corrected errors are not reported via machine check exceptions. Some corrected errors may be reported to software via error thresholding (see 2.16.1.7 [Error Thresholding]).

The implications of these categories of errors are:

1. Uncorrected error; hardware did not deal with the problem.
 - Operationally (error handling), action required, because program flow is affected.
 - Diagnostically (fault management), software may collect information to determine if any components should be de-configured or serviced.
2. Deferred error; hardware partially dealt with the problem via containment.
 - Operationally, action optional, because program flow has not been affected. However, steps may be taken by software to prevent access to the data in error.
 - Diagnostically, software may collect information to determine if any components should be de-configured or serviced.
3. Corrected error; hardware dealt with the problem.
 - Operationally, no action required, because program flow is unaffected.
 - Diagnostically, software may collect information to determine if any components should be de-configured or serviced.

Machine check conditions can be simulated to aid in debugging machine check handlers.

2.16.1.3 Error Detection, Action, Logging, and Reporting

Error detection is controlled by the MASK registers:
- Error detection for MCA controlled errors is enabled if not masked by MC*i*_CTL_MASK (see Table 58 [MCA Register Cross-Reference Table]).
- Error masking is performed regardless of MCA bank enablement in MCG_CTL (MSR0000_017B).

Error action refers to the hardware response to an error, aside from logging and reporting. Enablement of error action for each error is enumerated in the EAC (Error Action Condition) column of the error descriptions tables as follows:
- D: Detected. The error action is taken if the error is detected (i.e., not masked). These actions occur regardless of whether the MCA bank is enabled in MCG_CTL.
- E: Enabled. The error action is taken if the error is detected and the bank is enabled in MCG_CTL.

Error logging refers to the storing of information in the status registers, and is enabled if all of the following are true:
- Error detection is enabled.
- The MCA bank is enabled in MCG_CTL.

Error reporting refers to active notification of errors to software via machine check exceptions, and is enabled if all of the following are true:

- Error logging is enabled.
- The corresponding enable bit for the error in MC*i*_CTL is set to 1.

A machine check exception is generated if all the following are true:

- The error is uncorrected.
- The error is enabled for reporting.
- CR4.MCE is enabled.

Notes:

1. If CR4.MCE is clear, an error configured to cause a machine check exception causes a shutdown.
2. If error reporting is disabled, the setting of CR4.MCE has no effect.
3. If an uncorrected error is disabled from reporting, containment of the error and logging/reporting of subsequent errors may be affected. Therefore, unmasked uncorrected errors should be enabled for reporting for normal operation. Uncorrected errors should only be disabled from reporting for debug purposes.
4. Errors not associated with a specific core are reflected to core 0 of the compute unit. The error description tables identify which errors are associated or not associated with a specific core of the compute unit.

Throughout the MCA register descriptions, the terms “enabled” and “disabled” generally refer to reporting, and the terms “masked” and “unmasked” generally refer to logging, unless otherwise noted.

Some logged errors increment a counter in MCi_MISC, which may trigger an interrupt (see 2.16.1.7 [Error Thresholding]). Although no machine check exception is generated, these notifications can be viewed as “correctable machine check interrupts”.

For debug observability only, D18F3x180[ConvertUnCorToCorErrEn] can be used to log NB uncorrected errors as corrected errors.

2.16.1.3.1 MCA Conditions that Cause Shutdown

The following architectural conditions cause the processor to enter the Shutdown state; see section “Machine-Check Errors” in APM volume 2 for more detail; see 1.2 [Reference Documents]:

- Attempting to generate an MCE when machine check reporting is disabled at the system level (CR4.MCE == 0).
- Attempting to generate an MCE when a machine check is in progess on the same core (MSR0000_017A[MCIP] == 1).

The following non-architectural conditions cause the processor to enter the Shutdown state:

- EX “Retire dispatch queue parity” error. See Table 252 [MC5 Error Descriptions].
- EX “Mapper checkpoint array parity” error if UC == 1. See Table 252 [MC5 Error Descriptions].

2.16.1.3.2 Error Logging During Overflow

An error to be logged when the status register contains valid data can result in an overflow condition. During error overflow conditions, the new error may not be logged or an error which has already been logged in the status register may be overwritten. For the rules on error overflow, priority, and overwriting, see MSR0000_0401[Overflow].

Overflow alone does not indicate a shutdown condition. Uncorrected errors require software intervention. Therefore, when an uncorrected error cannot be logged, critical error information may have been lost, and

MCi_STATUS[PCC] may be set. If PCC is indicated, software should terminate system processing to prevent data corruption (see 2.16.1.6 [Handling Machine Check Exceptions]). If PCC is not indicated, any MCA data lost due to overflow was informational only and not critical to system hardware operation.

The following table indicates which errors are overwritten in the error status registers.

Table 59: Overwrite Priorities for All Banks

			Older Error					
			Uncorrected		Deferred		Corrected	
			Enabled	Disabled	Enabled	Disabled	Enabled	Disabled
Newer Error	Uncorrected	Enabled	-	Overwrite	Overwrite	Overwrite	Overwrite	Overwrite
		Disabled	-	-	Overwrite	Overwrite	Overwrite	Overwrite
	Deferred	Enabled	-	-	-	Overwrite	Overwrite	Overwrite
		Disabled	-	-	-	-	Overwrite	Overwrite
	Corrected	Enabled	-	-	-	-	-	Overwrite
		Disabled	-	-	-	-	-	-

2.16.1.4 MCA Initialization

The following initialization sequence must be followed:

- MCi_CTL_MASK registers (see Table 58 [MCA Register Cross-Reference Table] for list):
 - BIOS must initialize the mask registers to inhibit error detection prior to the initialization of MCi_CTL and MSR0000_017B.
 - BIOS must not clear MASK bits that are reset to 1.
- The MCi_CTL registers must be initialized by the operating system prior to enabling the error reporting banks in MCG_CTL.

If initializing after a cold reset (see D18F0x6C[ColdRstDet]), then BIOS must clear the MCi_STATUS MSRs. If initializing after a warm reset, then BIOS should check for valid MCA errors and if present save the status for later diagnostic use (see 2.16.1.6 [Handling Machine Check Exceptions]).

BIOS may initialize the MCA without setting CR4.MCE; this results in a system shutdown on any machine check which would have caused a machine check exception (followed by a reboot if configured in the chipset). Alternatively, BIOS that wishes to ensure continued operation in the event that a machine check occurs during boot may write MCG_CTL with all ones and write zeros into each MCi_CTL. With these settings, a machine check error results in MCi_STATUS being written without generating a machine check exception or a system shutdown. BIOS may then poll MCi_STATUS during critical sections of boot to ensure system integrity. Before passing control to the operating system, BIOS should restore the values of those registers to what the operating system is expecting. (Note that using MCi_CTL to disable error reporting on uncorrected errors may affect error containment; see 2.16.1.3 [Error Detection, Action, Logging, and Reporting].)

2.16.1.5 Error Code

The MCi_STATUS[ErrorCode] field contains information used to identify the logged error. Table 60 [Error Code Types] identifies how to decode ErrorCode. The MCi_STATUS[ErrorCodeExt] field contains detailed, model-specific information that is used to further narrow identification for error diagnosis, but not error handling by software; see 2.16.1.6 [Handling Machine Check Exceptions].

For a given error-reporting bank, Error Code Type is used in conjunction with the Extended Error Code (MCi_STATUS[ErrorCodeExt]) to uniquely identify the Error Type; the value of ErrorCodeExt is unique

within Error Code Type. Details for each Error Type are described in the tables accompanying the MCi_STATUS register for each bank.

- MC0 (LS); Table 240 [MC0 Error Signatures].
- MC1 (IF); Table 243 [MC1 Error Signatures].
- MC2 (CU); Table 247 [MC2 Error Signatures].
- MC4 (NB); Table 250 [MC4 Error Signatures, Part 1] and Table 251 [MC4 Error Signatures, Part 2].
- MC5 (EX); Table 253 [MC5 Error Signatures].
- MC6 (FP); Table 256 [MC6 Error Signatures].

Table 60: Error Code Types

Error Code	Error Code Type	Description
0000 0000 0001 TTLL	TLB	TT = Transaction Type LL = Cache Level
0000 0001 RRRR TTLL	Memory	Errors in the cache hierarchy (not in NB) RRRR = Memory Transaction Type TT = Transaction Type LL = Cache Level
0000 1PPT RRRR IILL	Bus	General bus errors including link and DRAM PP = Participation Processor T = Timeout RRRR = Memory Transaction Type II = Memory or IO LL = Cache Level
0000 01UU 0000 0000	Internal Unclassified	Internal unclassified errors UU = Internal Error Type

Table 61: Error Codes: Transaction Type (TT)

TT	Transaction Type
00	Instr: Instruction
01	Data
10	Gen: Generic
11	Reserved

Table 62: Error Codes: Cache Level (LL)

LL	Cache Level
00	Reserved
01	L1: Level 1
10	L2: Level 2
11	LG: Generic

Table 63: Error Codes: Memory Transaction Type (RRRR)

RRRR	Memory Transaction Type
0000	Gen: Generic. Includes scrub errors.
0001	RD: Generic Read

Table 63: Error Codes: Memory Transaction Type (RRRR) (Continued)

RRRR	Memory Transaction Type (Continued)
0010	WR: Generic Write
0011	DRD: Data Read
0100	DWR: Data Write
0101	IRD: Instruction Fetch
0110	Prefetch
0111	Evict
1000	Snoop (Probe)

Table 64: Error Codes: Participation Processor (PP)

PP	Participation Processor
00	SRC: Local node originated the request
01	RES: Local node responded to the request
10	OBS: Local node observed the error as a third party
11	Generic

Table 65: Error Codes: Memory or IO (II)

II	Memory or IO
00	Mem: Memory Access
01	Reserved
10	IO: IO Access
11	Gen: Generic

Table 66: Error Codes: Internal Error Type (UU)

UU	Internal Error Type
00	Reserved
01	Reserved
10	HWA: Hardware Assertion
11	Reserved

2.16.1.6 Handling Machine Check Exceptions

A machine check handler is invoked to handle an exception for a particular core. Because MCA registers are generally not shared among cores, the handler does not need to coordinate register usage with handler instances on other cores. Those few MCA registers which are shared are noted in the register description. (See also 2.4.2.1 [Registers Shared by Cores in a Compute Unit].)

For access to the NB MCA registers, D18F3x44[NbMcaToMstCpuEn] allows a single core (the NBC) to access the registers through MSR space without contention from other cores. This organization of registers on a per core basis allows independent execution, simplifies exception handling, and reduces the number of conditions which are globally fatal.

At a minimum, the machine check handler must be capable of logging error information for later examination.

The handler should log as much information as is needed to diagnose the error.

More thorough exception handler implementations can analyze errors to determine if each error is recoverable by software. If a recoverable error is identified, the exception handler can attempt to correct the error and restart the interrupted program. An error may not be recoverable for the process or virtual machine it directly affects, but may be containable, so that other processes or virtual machines in the system are unaffected and system operation is recovered; see 2.16.1.6.1 [Differentiation Between System-Fatal and Process-Fatal Errors].

Machine check exception handlers that attempt to recover must be thorough in their analysis and the corrective actions they take. The following guidelines should be used when writing such a handler:

- Data collection:
 - All status registers in all error reporting banks must be examined to identify the cause of the machine check exception.
 - Read MSR0000_0179[Count] to determine the number of status registers visible to the core. The status registers are numbered from 0 to one less than the value found in MSR0000_0179[Count]. For example, if the Count field indicates five status registers are supported, they are numbered MC0_STATUS to MC4_STATUS. These are generically referred to as MC*i*_STATUS.
 - Check the valid bit in each status register (MC*i*_STATUS[Val]). The remainder of the status register should be examined only when its valid bit is set.
 - When identifying the error condition and determining how to handle the error, portable exception handlers should examine the following MC*i*_STATUS fields: ErrorCode, UC, PCC, CECC, UECC, Deferred, Poison. The expected settings of these and other fields in MC*i*_STATUS are identified in the error signatures tables which accompany the descriptions of each MCA status register. See 2.16.1.5 [Error Code] for a discussion of error codes and pointers to the error signatures tables.
 - MC*i*_STATUS[ErrorCodeExt] should generally not be used by portable code to identify the error condition because it is model specific. ErrorCodeExt is useful in determining the error subtype for root cause analysis.
 - Error handlers should collect all available MCA information (status register, address register, miscellaneous register, etc.), but should only interrogate details to the level which affects their actions. Lower level details may be useful for diagnosis and root cause analysis, but not for error handling.
- Recovery:
 - Check the valid MC*i*_STATUS registers to see if error recovery is possible.
 - Error recovery is not possible when the processor context corrupt indicator (MC*i*_STATUS[PCC]) is set to 1.
 - The error overflow status indicator (MCi_STATUS[Overflow]) does not indicate whether error recovery is possible. See 2.16.1.3.2 [Error Logging During Overflow].
 - If error recovery is not possible, the handler should log the error information and return to the operating system for system termination.
 - Check MC*i*_STATUS[UC] to see if the processor corrected the error. If UC is set, the processor did not correct the error, and the exception handler must correct the error prior to attempting to restart the interrupted program. If the handler cannot correct the error, it should log the error information and return to the operating system. If the error affects only process data, it may be possible to terminate only the affected process or virtual machine. If the error affects processor state, continued use of that processor should not occur. See individual error descriptions for further guidance.
 - If MSR0000_017A[RIPV] is set, the interrupted program can be restarted reliably at the instruction pointer address pushed onto the exception handler stack if any uncorrected error has been corrected by software. If RIPV is clear, the interrupted program cannot be restarted reliably, although it may be possible to restart it for debugging purposes. As long as PCC is clear, it may be possible to terminate only the affected process or virtual machine.
 - When logging errors, particularly those that are not recoverable, check MSR0000_017A[EIPV] to see if

the instruction pointer address pushed onto the exception handler stack is related to the machine check. If EIPV is clear, the address is not ensured to be related to the error.
 - See 2.16.1.6.1 [Differentiation Between System-Fatal and Process-Fatal Errors] for more explanation on the relationship between PCC, RIPV, and EIPV.
- Exit:
 - When an exception handler is able to successfully log an error condition, clear the MC*i*_STATUS registers prior to exiting the machine check handler. Software is responsible for clearing at least MC*i*_STATUS[Val].
 - Prior to exiting the machine check handler, be sure to clear MSR0000_017A[MCIP]. MCIP indicates that a machine check exception is in progress. If this bit is set when another machine check exception occurs in the same core, the processor enters the shutdown state.

Additional machine check handler portability can be added by having the handler use the CPUID instruction to identify the processor and its capabilities. Implementation specific software can be added to the machine check exception handler based on the processor information reported by CPUID.

In cases where Sync flood is the recommended response to a particular error, a machine check exception cannot be used in lieu of the Sync flood to stop the propagation of potentially bad data.

2.16.1.6.1 Differentiation Between System-Fatal and Process-Fatal Errors

The bits MC*i*_STATUS[PCC], MSR0000_017A[RIPV], and MSR0000_017A[EIPV] form a hierarchy, used by software to determine the degree of corruption and recoverability in the system.
Table 67 shows how these bits are interpreted.

Table 67: Error Scope Hierarchy

PCC	UC	RIPV	EIPV	Deferred	Poison	Comments
1	1	-	-	-	-	**System fatal error.** Signaled via machine check exception, action required. Error has corrupted system state (PCC == 1). The error is fatal to the system and the system processing must be terminated.
0	1	1	1	-	-	

Table 67: Error Scope Hierarchy (Continued)

PCC	UC	RIPV	EIPV	Deferred	Poison	Comments
0	1	0	-	-	0/1	**Hardware uncorrected, software containable error.** Signaled via machine check exception, action required. The error is confined to the process, however the process cannot be restarted even if the uncorrected error is corrected by software. Poison=1; the error is due to consumption of poison data. If the affected process or virtual machine is terminated, the system may continue operation.
0	0	-	-	1	0	**Deferred error**. Action optional. A latent error has been discovered, but not yet consumed; a machine check exception is generated if the affected data is consumed. Error handling software may attempt to correct this data error, or prevent access by processes which map the data, or make the physical resource containing the data inaccessible. Note: May be detected on a demand access or a scrub access.
0	0	-	-	0	0	**Corrected error**. Signaled via error thresholding mechanisms (2.16.1.7 [Error Thresholding]); no action required.

2.16.1.7 Error Thresholding

For some types of errors, the hardware maintains counts of the number of errors. When the counter reaches a programmable threshold, an event may optionally be triggered to signal software. This is known as error thresholding. The primary purpose of error thresholding is to help software recognize an excessive rate of errors, which may indicate marginal or failing hardware. This information can be used to make decisions about deconfiguring hardware or scheduling service actions. Counts are incremented for corrected, deferred, and uncorrected errors.

The error thresholding hardware counts only the number of errors; it is up to software to track the errors reported over time in order to determine the rate of errors. Thresholding gives error counts on groups of resources. In order to make decisions on individual resources, a finer granularity of error information, such as MCA information for specific errors, must be utilized in order to obtain more accurate counts and to limit the scope of actions to affected hardware.

Thresholding is performed for “Error Threshold Groups” identified in the list below. For all error threshold groups, some number of corrected errors is expected and normal. There are numerous factors influencing error rates, including temperature, voltage, operating speed, and geographic location. In order to accommodate the various factors, including software latency to respond and track the error thresholding, additional guardband above the normal rates is recommended before error rates are considered abnormal for purposes of hardware action.

The {MC0, MC1, MC2, MC5} error thresholding banks maintains counters, but do not provide interrupts when the threshold is reached; these counters must be polled.

Error thresholding groups:
- MC0
 - MC0 errors are counted and polled via MSR0000_0403.
 - MC0 errors are listed in Table 239 [MC0 Error Descriptions].
- MC1
 - MC1 errors are counted and polled via MSR0000_0407.
 - MC1 errors are listed in Table 242 [MC1 Error Descriptions].
- MC2
 - MC2 errors are counted and polled via MSR0000_040B.
 - MC2 errors are listed in Table 246 [MC2 Error Descriptions].
- Links (MC4)
 - Link errors are counted and polled or reported via MSRC000_0408.
 - Link errors are the errors listed in Table 250 [MC4 Error Signatures, Part 1] as “L” (Cache) in the ETG (Error Threshold Group) column.
 - For a link exhibiting excessive errors, it may be possible to reduce errors by lowering the link frequency or reducing the link width (if a bad lane can be avoided). See 2.11 [Root Complex] for details and restrictions on configuring links.
- MC5
 - MC5 errors are counted and polled via MSR0000_0417.
 - MC5 errors are listed in Table 253 [MC5 Error Signatures].

In rare circumstances, such as two simultaneous errors in the same error thresholding group, it is possible for one error not to increment the counter. In these conditions, MCi_STATUS[Overflow] may indicate that an overflow occurred, but the error counter may only indicate one error.

2.16.1.8 Error Diagnosis

This section describes generalized information and algorithms for diagnosing errors. The primary goal of diagnosis is to identify the failing component for repair purposes. The secondary goal is to identify the smallest possible sub-component for deallocation, deconfiguration, or design/manufacturing root cause analysis.

Indictment means identifying the part in error. The simplest form of indictment is *self-indictment*, where the bank reporting the error is the faulty unit. The next simplest form of indictment is *eyewitness indictment*, where the faulty unit is not the bank reporting the error, but is identified unambiguously. Both of these forms can be considered direct indictment; the information for indictment is contained in the MCA error information. If an error is not directly indicted, then identifying the faulty unit is more difficult and may not be an explicit part of the error log.

In general, an address logged in the MCA is useful for direct indictment only if the address identifies a physical location in error, such as a cache index. Logical addresses, while identifying the data, do not identify the location of the data.

If possible, physical storage locations in caches should be checked to determine whether the error is a soft error (a temporary upset of the stored value) or a hard fault (malfunctioning hardware). A location which has had a soft error can be corrected by writing a new value to the location; a reread of the location should see the new value. Hard faults cannot be corrected by writing a new value; the hardware persistently returns the previous value. If such checking is not possible, a grossly simplifying assumption can be made that uncorrected errors are hard and corrected errors are soft. Repeated corrected errors from the same location are an indication that the fault is actually hard.

Determining whether corrected errors represent a hard fault or a soft error requires understanding the access

patterns and any attempts to correct the faulty data in place. An attempt to correct the data in place creates two *epochs*, one before the correction event, and one after. If an error is seen at the same location in two different epochs (especially back-to-back epochs), it is more likely that the cause is a hard fault, since the error has persisted or repeated through an in place correction. The more epochs in which an error is seen, the higher the likelihood of it being caused by a hard fault.

As an example, consider a corrected error found during a read from DRAM. If the DRAM redirect scrubber is enabled, the data in error is corrected in place, and this event conceptually creates a new epoch. If the original fault was due to a soft error, a read of the same data in the new epoch should not encounter a data error. If the original fault was due to a hard fault (e.g., a stuck bit), a read of the data in the new epoch likely results in another corrected or uncorrected error.

There are numerous correction events that can be used to separate time periods into epochs. These include DRAM redirect scrubs, DRAM sequential scrubs, cache scrubs, cache writes, cache flushes, resets, and others.

2.16.1.8.1 Common Diagnosis Information

A common set of diagnosis information is useful for many problems. Table 68 indicates the minimum set of generally useful diagnostic information that should be collected by software, unless the specifics of the problem are known to be narrower, based on the error code or other information.

It is useful to collect configuration information to ensure that the behavior is not caused by misconfiguration.

Table 68: Registers Commonly Used for Diagnosis

MCA Bank	Status	Configuration
MC0	MSR0000_0401 MSR0000_0402 MSR0000_0403	MSR0000_0400 MSRC001_1022 MSRC001_0044
MC1	MSR0000_0405 MSR0000_0406 MSR0000_0407	MSR0000_0404
MC2	MSR0000_0409 MSR0000_040A MSR0000_040B	MSR0000_0408 MSRC001_0046 MSRC001_1023
MC3	Reserved	Reserved

Table 68: Registers Commonly Used for Diagnosis (Continued)

MCA Bank	Status	Configuration
MC4	MSR0000_0411 MSR0000_0412 MSR0000_0413 MSRC000_0408 D18F3x54 D18F2xAC	MSR0000_0410 MSRC001_0048 D18F3x40 D18F3x44 D18F3xE8 MSRC001_001F D18F3x180 D0F0xBC_xD820_0CE4
MC5	MSR0000_0415 MSR0000_0416 MSR0000_0417	MSR0000_0414
MC6	MSR0000_0419 MSR0000_041A MSR0000_041B	MSR0000_0418 MSRC001_004A

If examining MCA registers after startup, determine the cause of the startup:
- INIT; D18F0x6C[InitDet].
- Cold reset; D18F0x6C[ColdRstDet].
- Warm reset; if not INIT or Cold reset.

To see if a link failure occurred, examine D18F0x[E4,C4,A4,84][LinkFail]. If set, look for additional information:
- Receipt of a sync, such as during a Sync flood, saves a status of Sync Error in MC4_STATUS.
- CRC error saves a status of CRC Error in MC4_STATUS. See D18F0x[E4,C4,A4,84][CrcErr,CrcFloodEn].
- Link not present does not save status in MC4_STATUS. See D18F0x[E4,C4,A4,84][InitComplete].

Other registers may be needed depending on the specific error symptoms.

2.16.1.9 Deferred Errors and Data Poisoning

Deferred errors indicate error conditions which could not be corrected, but which require no action (i.e., action optional). Data poisoning marks data which has encountered an uncorrectable error, so that it can be tracked until it is consumed or discarded. Together, data poisoning and deferred errors provide an infrastructure for reducing the severity of errors and the number of system outages for some classes of errors.

Processor cores create poison data and deferred errors as identified in the Error Signatures tables. When poison data or data with an uncorrectable ECC error is consumed, a machine check exception for an uncorrected error is signaled (MCi_STATUS[UC]). If the data is poison, MCi_STATUS[Poison] is also set. The NB converts any poison data sent from the core to a machine check exception with error type Compute Unit Data Error. To understand the cause of a machine check exception due to Compute Unit Data Error, examine the core MCA status registers for deferred errors.

The deferred error is logged in the MCA registers for diagnostic purposes at the time the error is discovered and the data is poisoned. This deferred error can help identify the source of the error. The deferred error is logged independently of any associated poison data machine check. For example, it is possible for a cache eviction to result in a deferred error associated with the cache, and a corresponding machine check exception from the NB which receives the data.

2.17 FCH

The processor contains an integrated FCH. The FCH supports the following interfaces:
- Universal Serial Bus (USB) versions 2.0, and 3.0
- Serial ATA revision 3.0
- Secure Digital (SD)
- System Management Bus (SMBus)
- Low Pin Count (LPC) bus and SPI interface
- UART
- I^2C
- Serial IRQ
- Serial Peripheral Interface (SPI) ROM
- Advanced Configuration and Power Interface (ACPI)

The FCH also comprises the following functions:
- Real-Time Clock (RTC)
- Programmable Interrupt Controller (PIC)
- System clock generation
- System Management Interrupt (SMI)
- General-Purpose IO (GPIO)
- Power management
- Watchdog Timer (WDT)

2.17.1 MMIO Programming for Legacy Devices

The legacy devices LPC, IOAPIC, ACPI and Watchdog Timer require the base address of the Memory Mapped IO registers to be assigned before these logic blocks are accessed. The Memory Mapped IO register base address and its entire range should be mapped to a non-posted memory region by programming the CPU register.

2.17.2 USB Controllers

This processor supports one USB 2.0 EHCI controllers and one USB 3.0 xHCI controller. USB 2.0 HUBs are added between the USB2.0 PHYs and the EHCI controllers, so that both USB2.0 and USB1.1 devices can be supported. Each EHCI controller has 2 ports. Port 0 connects to the HUB and port 1 is used as the debug port. The HUB has 4 USB ports. There are 4 USB3.0 ports supported.

The USB2.0 EHCI controller supports USB2.0 ports and are mapping to port[3:0] through a dedicated HUB. The USB3.0 XHCI controller supports USB2.0 and USB3.0 ports connect ports[7:4].

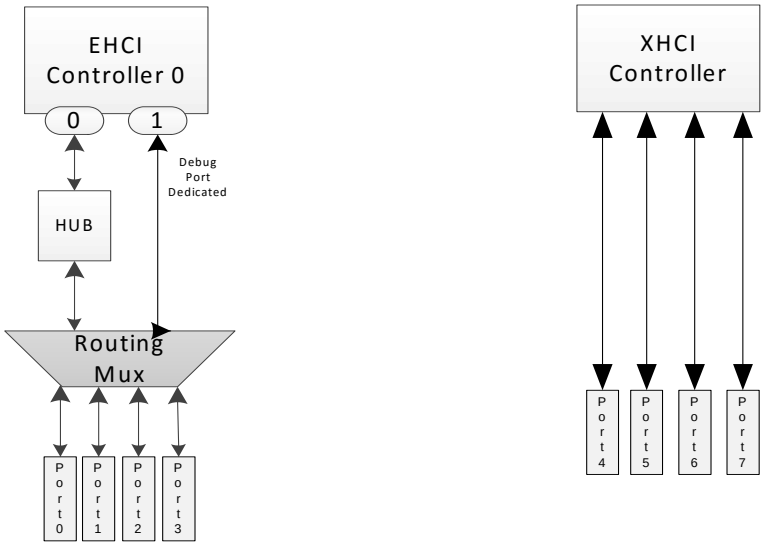


Figure 7: USB Port Configuration

Table 69: USB Port Mapping

Ports	Port type	Registers	Pins
3-0	USB2.0 Port	D12F0xXX, EHCI[3:0]xXX	USB_HSD[3:0]P/N
7-4	USB2.0 Port	D10F0xXX, XHCI_OPx4[3:0]4	USB_HSD[7:4]P/N
	USB3.0 Port	D10F0xXX, XHCI_OPx4[7:4]4	USB_SS_[3,0][RX,TX]P/N

Refer to the following for the USB controller register definitions:
- Section 3.26.4.1 [USB 2.0 (EHCI)] for the EHCI controllers.
- Section 3.26.4.2 [USB 3.0 (xHCI)] for the xHCI controller.

2.17.2.1 USB Power Management

USB power management functions are controlled by registers outside the USB controller registers spaces. See the following register definitions in 3.26.12 [Power Management (PM) Registers]:
- PMx80[UsbPeriodicalSetBmSts, Usb20SetBmSts]
- PMxED [USB Gating]
- PMxEF [USB Enable]
- PMxF0 [USB Control]

2.17.2.2 USB Interrupts

The interrupt mapping is specified by IOC00 [Pci_Intr_Index] and IOC01 [Pci_Intr_Data]. By default, USB interrupts are routed to PCI INTA#, INTB#, and INTC#.
For normal operation MSIs must be disabled in all OHCI and EHCI controllers.

2.17.2.3 Enabling the xHCI Controller

Software performs the following sequence to enable the xHCI controller:
1. Program PMxEF[Usb2EhciEnable] = 0.
2. Program PMxEF[PortRoutingSelect] = 1.
3. Program XHCI_PMx00[Xhci0Enable] = 1.

2.17.2.4 USB2.0 Controller PHY Configuration and Calibration

Software performs the following sequence to configure EHCI common PHY for each USB port to be used shortly after chip boot up and after USB2.0 EHCI controller initiated the calibration. The register settings have be to re-established to ensure USB2.0 PHY is properly calibrated. It is not needed after waking up from S3.
1. Program HSSLEW = 10b.
 A. Program EHCI1xB4[PortNumber] with the controller port number according to Table 70.
 B. Program EHCI1xB4[VControlModeSel] = 0110b.
 C. Program HSSLEW value at EHCI1xB4[1:0] = 10b.
 D. Program EHCI1xB4[VLoadB] = 1.
 E. Wait for EHCI1xB4[VBusy] = 0.
 F. Program EHCI1xB4[VLoadB] = 0.
 G. Wait for EHCI1xB4[VBusy] = 0.
 H. Program EHCI1xB4[VLoadB] = 1 to lock the PHY control interface.
2. Program EHCI1xD0[BgAdj] = 7.
3. Program EHCI1xC4[IRefAdj] = 2.
4. Program EHCI1xC4[XRefAdj] = 2.
5. Program EHCI1xC4[PVI] = 1.
6. Program EHCI1xC4[CPAdj] = 1.
7. Program EHCI1xC4[DLLControl] = 90h.
8. Program EHCI1xD4[PllFilter] = 1.
9. Program EHCI1xD4[CalEnable] = 0.
10. Wait for 200 ns.
11. Program EHCI1xD4[CalEnable] = 1.
12. Wait for 400 ns.
13. Program EHCI1xD4[CalEnable] = 0.

2.17.2.5 USB2.0 Controller Common PHY Impedance Calibration

The following sequence enables USB2.0 common PHY impedance calibration.
 A. Program EHCI1xC0[NewCalBus[6:3]] = 3.
 B. Program EHCI1xC0[AddToCommonCalibration] = 1 and EHCI1xC0[UseCommonCalibration] = 0 to have NewCaluBus replace ComCalBus for calibration.

2.17.2.6 USB2.0 Controller ISO Device CRC False Error Detection

Software performs the following sequence to program CDR phase shift limit for each USB port to be used:
1. Program EHCI1xB4[PortNumber] with the controller port number according to Table 70.
2. Program EHCI1xB4[VControlModeSel] = 0111b.
3. Program EHCI1xB4[2:0] = 101b.
4. Program EHCI1xB4[VLoadB] = 1.
5. Wait for EHCI1xB4[VBusy] = 0.
6. Program EHCI1xB4[VLoadB] = 0.
7. Wait for EHCI1xB4[VBusy] = 0.

8. Program EHCI1xB4[VLoadB] = 1 to lock the PHY control interface.

Table 70: USB Port to EHCI1xB4[PortNumber] Mapping

USB Port	Software selects by programming
0	EHCI1xB4[PortNumber]=0h
1	EHCI1xB4[PortNumber]=1h
2	EHCI1xB4[PortNumber]=2h
3	EHCI1xB4[PortNumber]=3h

2.17.2.7 xHC USB2.0 Common PHY Calibration

The following programming sequence is done after processor boot and after the USB2.0 EHCI/XHCI controller initiates the phy calibration. These register settings must also be reprogrammed after S4 and S5.

1. Program HSSLEW = 10b.
 A. Program D10F0x4C_x4000_0000[PortNumber] with the controller port number according to Table 69.
 B. Program D10F0x4C_x4000_0000[VControlModeSel] = 0110b.
 C. Program HSSLEW value at D10F0x4C_x4000_0000[1:0] = 10b.
 D. Program D10F0x4C_x4000_0000[VLoadB] = 1.
 E. Wait for D10F0x4C_x4000_0000[VBusy] = 0.
 F. Program D10F0x4C_x4000_0000[VLoadB] = 0.
 G. Wait for D10F0x4C_x4000_0000[VBusy] = 0.
 H. Program D10F0x4C_x4000_0000[VLoadB] = 1 to lock the PHY control interface.
2. Program D10F0x4C_x4000_0060[BgAdj] = 7.
3. Program D10F0x4C_x4000_001C[IRefAdj] = 2.
4. Program D10F0x4C_x4000_001C[XRefAdj] = 2.
5. Program D10F0x4C_x4000_001C[PVI] = 1.
6. Program D10F0x4C_x4000_001C[CPAdj] = 1.
7. Program D10F0x4C_x4000_001C[DllControl] = 90h.
8. Program D10F0x4C_x4000_0064[PllFilter] = 1.
9. Program D10F0x4C_x4000_0064[CalEnable] = 0.
10. Wait for 200 ns.
11. Program D10F0x4C_x4000_0064[CalEnable] = 1.
12. Wait for 400 ns.
13. Program D10F0x4C_x4000_0064[CalEnable] = 0.

2.17.2.8 xHC USB2.0 Port Drive Strength Adjustment

The drive strength is auto-calibrated on power up. If the calibrated values need to be adjusted, use the following procedure:

1. Program D10F0x4C_x4000_0000[PortNumber] with the controller port number according to Table 69.
2. Program D10F0x4C_x4000_0000[VControlModeSel] = 1000b to select Group 8.
3. Program D10F0x4C_x4000_0000[VControl[4]] = 0b and D10F0x4C_x4000_0000[VControl[5]] = 1b to use (Actual+Adjusted) CalAmp value for drive strength.
4. Program D10F0x4C_x4000_0000[VControl[3:0]] with desired drive strength adjustment.
 a. D10F0x4C_x4000_0000[VControl[3]] is the sign bit. 1=Negative offset. 0=Positive offset.
5. Program D10F0x4C_x4000_0000[VLoadB] = 1.
6. Wait for D10F0x4C_x4000_0000[VBusy] = 0.
7. Program D10F0x4C_x4000_0000[VLoadB] = 0.

8. Wait for D10F0x4C_x4000_0000[VBusy] = 0.
9. Program D10F0x4C_x4000_0000[VLoadB] = 1 to lock the phy control interface.

2.17.2.9 USB3.0 PHY Auto-Calibration Enablement

BIOS enables USB3.0 PHY auto-calibration as follows:
1. Program XHCI_PMx8C[PllVcoTune[3:0]] = 0011b.
2. Program XHCI_PMx8C[CrPllCalibEn] = 1.

2.17.2.10 USB3.0 Port LFPS Differential Detection Threshold Adjustment

BIOS sets different threshold values according to different power supplies:
1. If the platform nominal power supply is 0.95v, then BIOS programs SPHY Port[3:0] Test Control 0 XHCI_PMx4C_x2[C,8,4,0]0[RxLfpsDetTh[2:0]] = 4.
2. If the platform nominal power supply is 1.05v, then BIOS programs SPHY Port[3:0] Test Control 0 XHCI_PMx4C_x2[C,8,4,0]0[RxLfpsDetTh[2:0]] = 3.

2.17.2.11 xHCI ISO Device CRC False Error Detection

Software performs the following sequence to program CDR phase shift limit for each USB port to be used:
1. Program D10F0x4C_x4000_0000[PortNumber] with the controller port number according to Table 69.
2. Program D10F0x4C_x4000_0000[VControlModeSel] = 0111b.
3. Program D10F0x4C_x4000_0000[2:0] = 101b.
4. Program D10F0x4C_x4000_0000[VLoadB] = 1.
5. Wait for D10F0x4C_x4000_0000[VBusy] = 0.
6. Program D10F0x4C_x4000_0000[VLoadB] = 0.
7. Wait for D10F0x4C_x4000_0000[VBusy] = 0.
8. Program D10F0x4C_x4000_0000[VLoadB] = 1 to lock the phy control interface.

2.17.2.12 xHCI PHY Clock Gating

The following programming sequence configures the clock gating in the xHCI PHY. This must be performed for each port after enabling the xHCI controller and needs to be restored after all power state resumes:
1. Program D10F0x4C_x4000_0000[VLoadB] = 1.
2. Program D10F0x4C_x4000_0000[PortNumber] with the controller port number according to Table 69.
3. Program D10F0x4C_x4000_0000[VControlModeSel] = 0011b.
4. Program D10F0x4C_x4000_0000[2] = 1.
5. Read D10F0x4C_x4000_0000[VBusy] to ensure it is 0.
6. Program D10F0x4C_x4000_0000[VLoadB] = 0.
7. Wait for D10F0x4C_x4000_0000[VBusy] == 0.
8. Program D10F0x4C_x4000_0000[VLoadB] = 1 to lock the phy control interface.

2.17.2.13 xHCI Firmware Preload

Software performs the following programming sequence for firmware preload. Firmware preload should be executed before BIOS deasserts XHCI_PMx00[U3CoreReset].
1. Enable firmware preload by programming XHCI_PMx00[FwLoadMode] = 1.
2. Program XHCI_PMx04[XhciFwPreloadType] with 1 for boot RAM preload and 0 for instruction RAM preload.
3. Program XHCI_PMx04[XhciFwRomAddr] with the offset of the application firmware in the external ROM.
4. Program XHCI_PMx08[XhciFwRamAddr] = 0.

5. Program XHCI_PMx08[XhciFwSize] = 8000h.
6. Program XHCI_PMx00[FwPreloadStart] = 1 to start firmware preload.
7. Wait for XHCI_PMx00[FwPreloadComplete] == 1.
8. Program XHCI_PMx00[FwPreloadStart] = 0.
9. Program XHCI_PMx00[U3CoreReset] = 0 to de-assert xHC reset.

2.17.2.14 xHCI Clear Pending PME on Sx State Entry

During S3 and S4 entry with wake from Sx state enabled, if the xHCI controller received a wake event before the system shutdown into Sx state is completed, D10F0x54[PmeStatus] may remain set when the system enters into Sx state. This prevents subsequent wake events from being propagated to the ACPI controller. BIOS should write 1 to clear D10F0x54[PmeStatus] if D10F0x54[PmeStatus] is 1 and ACPI GEvent status bit is clear on entry into S3/S4 state.

2.17.3 SATA

2.17.3.1 SATA Operating Mode

The SATA host controller can operate in the following modes:
- IDE mode.
- AHCI mode.

Software programs the subclass code and programming interface register to enable the SATA controller as IDE controller or AHCI controller.
1. Program D11F0x40[SubclassCodeWriteEnable] = 1.
2. Program D11F0x08[SubclassCode] and D11F0x08[ProgramIF[7:0]] according to Table 271 [SATA Controller Subclass Code and ProgramIF Settings].
3. Program D11F0x40[SubclassCodeWriteEnable] = 0.

2.17.3.2 SATA Drive Detection in IDE Mode

The following sequence should be included in the SBIOS drive identification loop for SATA drive detection in IDE mode.

1. If any of the SATA port status registers at SATAx1[A,2]8[DET] == 03h, program IDE[1:0]x06[DriveHead] = A0h for the corresponding port and go to Step 2.
 Else, no SATA drives are attached; exit the detection loop program.
2. If (IDE[1:0]x06[DriveHead] == A0h) && (IDE[1:0]x07[7] == 0) && (IDE[1:0]x07[3] == 0) for a port, the SATA device on that port is ready. IDE[1:0]x07[7] means BSY status, IDE[1:0]x07[3] means DRQ status.
 Else, loop until 30 second time-out; there is no SATA device attached if a time-out occurs.
 Note: Most drives do not need the 30 second time-out. The 30 second time-out is only needed for some particularly large capacity SATA drives which require a longer spin up time during a cold boot.

2.17.3.3 SATA PHY Auto-Calibration Enablement

BIOS enables SATA PHY auto-calibration as follows:
1. Program D11F0x88[PllVcoTune] = 0011b.
2. Program PMxDC[PllCalibEn] = 1.
3. Reset PHY:
 A. Program D11F0x84[RSTB] = 0 to assert reset.
 B. Wait for 100 us.

C. Program D11F0x84[RSTB] = 1 to release phy reset.

4. Issue AHCI reset to reset the host controller.
 A. Program SATAx04[HR] = 1. Hardware clears this bit after AHCI reset is completed.

2.17.3.4 SATA PHY Fine Tuning

The SBIOS should program the SATA controller in the sequence indicated below to fine tune the phy. Performing this procedure provides sufficient time for the SATA controllers to correctly complete SATA drive detection. The same procedure is required after the system resumes from the S3 state.

- Gen 3 settings:
 - Program D11F0x80[15:0] = 0130h to select Gen 3 for all ports.
 - Program D11F0x98[31:0] = 0040_F407h.
- Gen 2 settings:
 - Program D11F0x80[15:0] = 0120h to select Gen 2 for all ports.
 - Program D11F0x98[31:0] = 0040_3204h to fine tune the phy for Gen 2.
- Gen 1 settings:
 - Program D11F0x80[15:0] = 0110h to select Gen 1 for all ports.
 - Program D11F0x98[31:0] = 0040_3103h to fine tune the phy for Gen 1.
- Squelch detector settings:

2.17.3.5 SATA PHY Reference Clock Selection

SATA PHY reference clock can be programmed to 48 MHz Non-Spread clock or 100 MHz Spread clock. BIOS default is 100 MHz spread clock. 48 MHz or 100 MHz clock is selected by setting MISCx1C[SataRefClkSrc]. The following steps describes the programming required for each clock selection.

1. Program PHY to off-line.
 - SATAx12C[DET] = 4h // PORT 0 off-line.
 - SATAx1AC[DET] = 4h // PORT 1 off-line.
 - See SATAx1[A,2]C.
2. Program Clock source 48 MHz (Non Spread) or 100 Mhz (Spread).
 - MISCx1C[SataRefClkSrc] = 0 selects 48 MHz (Hardware Default); or
 - MISCx1C[SataRefClkSrc] = 1 selects 100 MHz (BIOS Default).
 - Note: On every Sx state entry, the MISCx1C[SataRefClkSrc] should be set to 0. On resume from Sx states, the register should be programmed to 1.
3. Select internal reference clock.
 - Program PMxDA[RefClkSel] = 1 to select internal reference clock.
4. Program PLL setting according to clock selection:
 - IF MISCx1C[SataRefClkSrc] == 0 THEN
 - Internal 48 MHz differential non-spread clock from pad.
 i. Program PMxDA[RefDivSel] = 0 to set PHY divider to divide by 1.
 ii. Program D11F0x8C[PLL_CLKF] = 7Dh to set PLL feedback clock divider value to 6 GHz/48 MHz = 7Dh.
 - ELSEIF MISCx1C[SataRefClkSrc] == 1 THEN
 - Internal 100 MHz spread clock.
 i. Program PMxDA[RefDivSel] = 2 to set PHY divider to divide by 4.
 ii. Program D11F0x8C[PLL_CLKF] = 3Ch to set PLL feedback clock divider value to 6 GHz/100 MHz = 3Ch.
5. Program PHY to on-line.
 - SATAx12C[DET] = 1h // PORT 0 on-line.

- SATAx1AC[DET] = 1h // PORT 1 on-line.

6. Reset PHY:
 A. Program D11F0x84[RSTB] = 0 to assert reset.
 B. Wait for 100 us.
 C. Program D11F0x84[RSTB] = 1 to release PHY reset.
7. Issue an AHCI reset to reset the host controller.
 A. Program SATAx04[HR] = 1. Hardware clears this bit after AHCI reset is complete.

2.17.3.6 SATA Power Management

2.17.3.6.1 SATA PHY Power Saving

The SATA PHY has different power saving states with the Active state consuming the most power. The table below summarizes the different controls during each power saving state:

Table 71: SATA PHY Power Saving States

Power Saving States	PMxDA[SetMaxGen2]	PMxDA[SataEnable]	SATAx1[9,1]8[ICC]	D11F0x40[SataPortDisable]
Active	0	1	1	0
Active with SetMaxGen2	1	1	1	0
Partial	X	1	2	0
Slumber	X	1	6	0
Lane Disabled	X	1	X	1
PHY Power Down	X	0	X	1

2.17.3.7 Enable Shadow Register Reload

When the default PHY settings are not optimal, they need to be adjusted by software. Once these settings are re-programmed, software shall program D11F0x84[S5ShadowLdDis] = 0 to turn on shadow register reloading. If the default PHY settings are optimal, D11F0x84[S5ShadowLdDis] should retain the value of 1 (default), thus disabling S5 Shadow Register reloading.

2.17.3.8 SATA Interrupt Handling

2.17.3.8.1 Line Interrupt

The SATA interrupt mapping is specified by IOC00 [Pci_Intr_Index] and IOC01 [Pci_Intr_Data].

2.17.3.8.2 MSI Message

The SATA controller supports message-based interrupts. When MSI is enabled and the SATA controller owns N ports, as specified by SATAx0C [Ports Implemented (PI)], D11F0x50[MMC] should be programmed according to platform configurations:
• If SATAx00[CCCS] == 1, D11F0x50[MMC] should be the minimum of 2M which satisfies 2M >= (N+1). For example: if BIOS knows that system has 2 SATA ports in AHCI mode (N=2). If SATAx00[CCCS] == 1, BIOS should program D11F0x50[MMC] = 2 to request 4 MSI interrupts.

The MSI capability pointer should be hidden when SATA subclass is configured in IDE mode (see 2.17.3.1 [SATA Operating Mode]).

2.17.3.9 Clear status of SATA PERR

BIOS should clear SATA PERR status soon after a cold boot, warm boot, any S3/S1 resume events, or an IOCF9 initiated reset outside a warm boot before any SATA activity is initiated.
- Write-1-to-clear SMIx3C[SataPerr].
- Write-1-to-clear SMIx84[RasEvent55].

2.17.4 LPC Bus Interface

BIOS should program D14F3xA0[SpiBaseAddr] with non-zero address to enable the MMIO access to SPI Flash control registers.

2.17.4.1 Enabling LPC DMA Function

If DMA is required for the LPC interface, program the registers as follows:
- Program D14F3x40[LegacyDmaEnable] = 1.
- Program D14F3x78[NoHog] = 1.
- Program PMx08[ArbDmaDis] = 1.
- Program D14F3x78[LDRQ1] = 1 and D14F3x78[LDRQ0] = 1.
- Program PMx04[LegacyDmaPrefetchEnhance] = 1 for non-DOS mode. Note: This bit should only be enabled in the ACPI method (called by the OS). This ensures that it is enabled only when the system is in Windows® mode. Under DOS mode, this feature may cause the undefined floppy drive behavior.

2.17.4.2 Enabling SPI 100

To enable the support for SPI 100 MHz speed, software programs SPIx20[UseSpi100] = 1. SPI 100 should be enabled after Auto ROM sizing has completed (Auto ROM sizing is done by the software during the initial boot when the system power state transitions from G3->S5->S0). The actual read speed also depends on the settings at SPIx22 [SPI100 Speed Config].

2.17.4.3 Enabling LPC_SMI Function

GPIO pins can be used to support LPC_SMI functionality. Program the appropriate pin that is selected to connect to the LPC Device for SMI function, and enable the GPIO function of the pin as shown in Table 72. IOMUXx56 [LPC_SMI_L_AGPIO86] should not be used to support the LPC_SMI function.
Please refer to the Functional Data Sheet for the AMD Socket used for detail description of the pins. See section 1.2 [Reference Documents] for Functional Data Sheet information.

To configure LPC_SMI functionality, the following must be done:
- Configure the AGPIO pin to be used by setting the correct IOMUX (see Table 72).
- Program the “SMI Control” and “SciMap” registers correspoinding to the pin selected for LPC_SMI.

Table 72: LPC_SMI Pins

Pin name	GPIO Setting
AGPIO3	IOMUXx03[AGPIO3] = 00b
AGPIO6/LDT_RST_L	IOMUXx06[AGPIO6_LDT_RST_L] = 00b
IR_RX1/AGPIO15	IOMUXx0F[IR_RX1_AGPIO15] = 01b

Table 72: LPC_SMI Pins (Continued)

Pin name	GPIO Setting
LPC_PD_L/AGPIO21	IOMUXx15[LPC_PD_L_AGPIO21] = 01b
LPC_PME_L/AGPIO22	IOMUXx16[LPC_PME_L_AGPIO22] = 01b
USB_OC3_L/T DO/AGPIO24	IOMUXx18[TDO_USB_OC3_L_AGPIO24] = 01b

2.17.4.4 Serial Peripheral Interface (SPI) ROM

SPI memory can be protected from being written or read from by using the following sequence. This is optional and used as required.

2.17.4.4.1 Enable SPI ROM Protection

1. Program D14F3x[5C,58,54,50] [ROM Protect 3, 2, 1, 0] to enable protection for Read or Write memory accesses to SPI flash memory space. Up to four ranges of memory can be protected.
2. Program SPIx04 [SPI_RestrictedCmd] or SPIx08 [SPI_RestrictedCmd2] with SPI commands that are required before doing Write accesses to SPI ROM. Use commands such as WR_EN, WR_Status and Erase when programing these restricted command registers.
3. Program SPIx00[SpiAccessMacRomEn] = 0 to protect the registers above from being re-programmed.

Now accesses are blocked by the SPI host preventing write access to SPI rom.

Notes:
Step 1 enables protection for direct memory access to SPI (Direct memory access is when Host access SPI memory using normal Read / Write commands these access are converted to SPI Read and Write commands by the SPI controller).
Step 2 enables protection for indirect memory access to the SPI controller. Indirect memory accesses are when the SPIx00 [SPI_Cntrl0] register is used to send SPI commands to the SPI ROM.
Step 3 By programming the SPIx00[SpiAccessMacRomEn] = 0, all SPI commands (defined in register SPIx14 [SPI_CmdValue1] and SPIx18 [SPI_CmdValue2]) and Restricted commands (defined in SPIx04 [SPI_RestrictedCmd] and SPIx08 [SPI_RestrictedCmd2]) are protected.

2.17.4.4.2 Disable SPI ROM Protection

1. Generate SMI. See 2.4.9.2.1 [SMM Overview].
2. Program D14F3x[5C,58,54,50][ReadProtect, WriteProtect] = 0 to un-protect Read or Write memory access SPI flash memory space.
3. Program SPIx00[SpiAccessMacRomEn] = 1 to unprotect the registers above from being re-programmed.
4. Exit SMI mode.

2.17.5 SD Controller

The SD controller is SD 3.0 compliant.

2.17.6 ASF Controller

The ASF controller can be used in a slave mode to support DASH clients. When the platform is configured for DASH the BIOS should at the minimum set the following registers to ensure the ASF is configured in slave mode and the control commands are defined properly in the ASF tables.

- When operating in Master mode, BIOS should program PMx00[AsfSmMasterEn] = 1. BIOS should program PMx00[AsfSmMasterEn] back to 0 after the master operations are done.
- When operating in Slave mode, BIOS should program PMx00[AsfSmMasterEn] = 0.

In addition to supporting remote DASH operations, BIOS should ensure that the ASF tables define the correct command codes and ASFx0E [RemoteCtrlAdr] has to be programmed with the same value reported in the ASF control data table.

The ASF controller can behave like a remote control device to accept commands from client side to do the reset/power down/power up/power cycle. BIOS reports those in the ASF control data table. The command and command data are defined in Table 73 [ASF Remote Control Commands].

Table 73: ASF Remote Control Commands

Remote Control	Control Command	Control data
Reset	50h	00h
Power Up	51h	00h
Power Down	52h	00h
Power Cycle	53h	00h

2.17.7 Integrated Micro-Controller

The FCH supports an integrated micro-controller (IMC).

The IMC can be enabled by programming PMxD6[ImcEnable].

2.17.8 On-Chip Clock Generator

There are two clocking modes in which this chip can be brought up based on the boot strap pin LPCCLK1. SBIOS should read MISCx80[ClkGenStrap] to determine clock generator mode. SBIOS should program MISCx40[OscClkSwitchEn] to 1 to select average 14 MHz OSC clock provided by internal PLL in both internal and external clocking mode.

2.17.8.1 Power Saving In Internal Clock Mode

The GPP_CLK_P/N pins are powered off when the processor is strapped to use an external clock and powered on when strapped to operate in internal clock mode. The GPP_CLK clocks are mapped to corresponding CLK_REQ# pins. The GPP_CLK0/1/2/3 mapping is defined at MISCx00[15:0]. The SLT_GFX_CLK mapping is defined at MISCx04[7:4]. In internal clock mode, a selected GPP_CLK can be powered off when the corresponding CLK_REQ# is asserted. See MISCx00 && MISCx04 for details.

Software can also program MISCx04[PCIE_RCLK_PowerDownEnable] = 1 to turn off the 100 MHz reference clock input buffer in internal clock generator mode for power savings.

2.17.8.2 Global A-Link / B-Link Clock Gating

Software programs MISCx2C[AlinkClkGateOffEn] = 1 and PMx04[ABLinkClkGateEn] = 1 to enable global A-link clock gate off function.
Software programs MISCx2C[BlinkClkGateOffEn] = 1 and PMx04[ABLinkClkGateEn] = 1 to enable global B-link clock gate off function.
PMx04[ABLinkClkGateEn] is a non-sticky bit should be programmed to 1 after PCI reset and S3/S4/S5 state if global A-link / B-link gating functions been enabled. MISCx2C[AlinkClkGateOffEn] and MISCx2C[Blink-

ClkGateOffEn] are sticky bits.

2.17.8.3 CG_PLL CMOS Clock Driver Setting for Power Saving

Software programs the following registers to select CMOS clock driver for CG1_PLL in internal & external clock mode for power saving:

A. Program MISCx1C[CgpllClkDriverUpdate] = 0.
B. Program MISCx1C[Cg1ClkDriverType] = 1011b and Program MISCx1C[Cg1RefClk48MHzDriverType] = 0010b.
C. Perform a software reset through IOCF9.

These settings only need to be applied once when the processor boots up from G3-S5->S0.

2.17.9 FCH Gasket

The FCH gasket (SBG) is responsible for converting between the protocols used by the interface and the A-link bridge interface. BIOS should program PMxE0 to set up base address before accessing AB configuration registers.

2.17.10 A-Link Bridge

The A-link bridge (AB) sits behind the FCH gasket.

2.17.10.1 Detection of Upstream Interrupts

BIOS should enable AB to detect upstream interrupts for the purpose of system management.

- Program ABx04_x94[MsiAddr[39:20]] with CPU interrupt delivery address [39:20].
- Program ABx04_x94[MsiAddrEn] = 1.

2.17.10.2 AB Memory Power Saving

The following sequence enables AB memory power saving:

A. Program MISCx68[ABBypassMemDsd] = 0 to enable AB memory DSD.
B. Program ABx04_x58[BlMemSDEn] = 0 to disable B-link memory shutdown.
C. Program ABx04_x58[AlMemSDEn] = 0 to disable A-link memory shutdown.

2.17.10.3 AB Internal Clock Gating

The following sequence enables AB internal clock gating:

A. Program ABx04_x54[BlClkGateDelay] = 10h to set the number of cycles to delay before gating B-link clocks.
B. Program ABx04_x10054[AlClkGateDelay] = 10h to set the number of cycles to delay before gating A-link clocks.
C. Program ABx04_x54[DbgClkGateEn] = 1 to disable debug only B-link clocks.
D. Program ABx04_x10054[AlClkGateEn] = 1 to enable medium grain A-link clock gating.
E. Program ABx04_x10054[DbgClkGateEn] = 1 to disable debug only A-link clocks.

2.17.10.4 AB 32/64-Byte DMA Write Enable

To enable AB 32-byte/64-byte DMA writes, program ABx04_x54[UpWr16BMode] = 0 and ABx04_x54[UpSWrByteCntSbgMode] = 1 to disable 16 byte write splitting and enable the SBG mode byte-count calculation logic. Software should also ensure that ABx04_x204[Dma16ByteMode] = 0.

2.17.11 Port Detection on XHCI Controller

2.17.12 AutoSize Function

The following steps enable the hardware to automatically save/restore the firmware location obtained by Auto-Size function:

1. Program D14F3xC8[AutoSizeStart] = 1 to start the hardware autosize Function.
2. Read the detected firmware location through D14F3xCC[AutoAddressSelect] and D14F3xCC[AutoRomAddr].
3. Program D14F3xCC[AutoSel] = 1.
4. Program PMxB9[AutoSizeDone_pmio] = 1.
5. Write the firmware location information obtained in Step 2 to AutoRomAddr registers D14F3xCC[AutoRomAddr].

2.17.13 FCH IP Address Mapping

There are several IPs embedded in FCH. Some of them are connected directly under PCI Bus, others are connected under AMBA APB bus.

The address mapping under APB slave list is as follows:

Table 74: Address Mapping Under APB Slave

Function Name	Address Mapping
AL2AHB	0xFEDC_ 0XXX
I^2C_0	0xFEDC_2XXX
I^2C_1	0xFEDC_3XXX
I^2C_2	0xFEDC_4XXX
I^2C_3	0xFEDC_5XXX
UART_0	0xFEDC_6XXX
DMA_0	0xFEDC_7XXX
UART_1	0xFEDC_8XXX
DMA_1	0xFEDC_9XXX

3 Registers

This section provides detailed field definitions for the core register sets in the processor.

3.1 Register Descriptions and Mnemonics

Each register in this document is referenced with a mnemonic. Each mnemonic is a concatenation of the register-space indicator and the offset of the register. Here are the mnemonics for the various register spaces:

- **IOXXX**: x86-defined input and output address space registers; XXX specifies the hexidecimal byte address of the IO instruction. This space includes IO-space configuration access registers, IOCF8 [IO-Space Configuration Address] and IOCFC [IO-Space Configuration Data Port], and the legacy block configuration registers. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.2 [IO Space Registers] and 3.26.1 [Legacy Block Configuration Registers (IO)].
- **APICXX0**: APIC memory-mapped registers; XX0 is the hexidecimal byte address offset from the base address. See 2.4.9.1.2 [APIC Register Space].
- **CPUID FnXXXX_XXXX_EiX[_xYYY]**: processor capabilities information returned by the CPUID instruction. See 3.19 [CPUID Instruction Registers]. Each core may only access this information for itself.
- **MSRXXXX_XXXX**: MSRs; XXXX_XXXX is the hexidecimal MSR number. This space is accessed through x86-defined RDMSR and WRMSR instructions. Unless otherwise specified there is one set of these registers Per-core. See 2.4.1 [Compute Unit].
- **DXFYxZZZ**: PCI-defined configuration space; X specifies the hexadecimal device number (this may be 1or 2 digits), Y specifies the function number, and ZZZ specifies the hexidecimal byte address (this may be 2 or 3 digits; e.g., D18F3x40 specifies the register at device 18h, function 3, and address 40h). See 2.7 [Configuration Space], for details about configuration space.
 - Some registers in D18F2xXXX have the _dct[0] mnemonic suffix. See 2.9.3 [DCT Configuration Registers].
 - Some registers in D0F2xXXX have the _L1i[3:0] mnemonic suffix. See 2.12.2.1 [IOMMU L1 Initialization].
- **IOMMUxX_XXXX**: IOMMU memory mapped registers; X_XXXX specifies the hexadecimal byte address offset (this may be 2 to 5 digits) from the base address register; The base address for this space is specified by D0F2x44 [IOMMU Base Address Low] and D0F2x48 [IOMMU Base Address High]. See 3.17 [IOMMU Memory Mapped Registers].
- **PMCxXXX**: core performance monitor events; XXX is the hexidecimal event counter number programmed into MSRC001_020[A,8,6,4,2,0][EventSelect]; See 2.6.1.1 [Core Performance Monitor Counters].
 - When PMCxXXX is followed by [z:y] then UnitMask[z:y] is being specified.
- **NBPMCxXXX**: NB performance monitor events; XXX is the hexadecimal event counter number programmed into MSRC001_024[6,4,2,0][EventSelect]; See 2.6.1.2 [NB Performance Monitor Counters].
 - When NBPMCxXXX is followed by [z:y] then UnitMask[z:y] is being specified.
- **ABxXX**: ALink Bridge registers; XX specifies the hexadecimal byte address offset from the base address. The base address for this space is specified by PMxE0 [ABRegBar]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.2 [AB Configuration Registers].
- **IDE[Y]xXX**: SATA controller IDE mode IO mapped registers; XX specifies the hexadecimal byte address offset from the base address; Y specifies primary drive (0) or secondary drive (1). The base address for this space is specified in Table 272 [IDE Compatibility Mode and Native Mode Address Mapping]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.3.2.1 [IDE Compatibility Mode and Native Mode Registers].
- **IDE_BMxXX**: SATA controller IDE mode IO mapped bus master registers; XX specifies the hexadecimal byte address offset from the base address. The base address for this space is specified by D11F0x20 [Bus Master Interface Register Base Address (BAR4)]. Unless otherwise specified, there is one set of these regis-

ters per node; the registers in a node are accessible to any core on that node. See 3.26.3.2.2 [IDE Bus Master Registers].

- **SATAxXXX**: SATA controller AHCI mode memory mapped registers; XXX specifies the hexadecimal byte address offset from the base address. The base address for this space is specified by D11F0x24 [AHCI Base Address (BAR5)]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.3.3 [SATA Memory Mapped AHCI Registers].
- **SATA_EMxXX**: SATA controller AHCI mode memory mapped enclosure buffer management registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.3.3.3 [Enclosure Buffer Management Registers].
- **EHCI[Y]xXX**: USB EHCI controller memory mapped registers; XX specifies the hexadecimal byte address offset from the base address; Y specifies the EHCI controller number (e.g., EHCI[1]x04 specifies EHCI controller 1 memory mapped control register offset 04). The base address for this space is specified by D12F0x10 [BAR_EHCI]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.4.1.2 [EHCI Memory Mapped IO Registers].
- **xHCI_PMxXX**: USB xHCI controller ACPI memory mapped registers; XX specifies the hexadecimal byte address offset from the base address. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.4.2.3 [xHCI Power Management Registers].
- **SDHCxXX**: Secure Digital host controller memory mapped registers; XX specifies the hexadecimal byte address offset from the base address. The base address for this space is specified by {D14F7x14 [Upper Base Address Reg 0], D14F7x10 [Base Address Reg 0]}. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.5.2 [SD Host Controller Configuration Registers (SDHC)].
- **ASFxXX**: ASF registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.6.2 [ASF (Alert Standard Format) Registers].
- **SMBUSxXX**: SMBus registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.6.3 [SMBus Registers].
- **IOAPICxXX**: IOAPIC registers; XX specifies the hexadecimal byte address offset from the base address. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.7 [IOAPIC Registers].
- **SPIxXX**: SPI memory mapped registers; XX specifies the hexadecimal byte address offset from the base address. The base address is specified by D14F3xA0 [SPI Base_Addr]. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.8.2 [SPI Registers].
- **HPETxXXX**: HPET registers; XXX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.9 [High Precision Event Timer (HPET) Registers].
- **MISCxXX**: ACPI miscellaneous control registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.10 [Miscellaneous (MISC) Registers].
- **GPIOxXX**: GPIO registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.11.1 [GPIO Registers].
- **IOMUXxXX**: IOMux registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.11.2 [IOMux Registers].
- **PMxXX**: power management registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that

node. See 3.26.12 [Power Management (PM) Registers].
- **PM2xXX**: power management block 2 registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.13 [Power Management Block 2 (PM2) Registers].
- **SMIxXX**: SMI registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.15 [SMI Registers].
- **WDTxXX**: Watchdog timer registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.16 [Watchdog Timer (WDT) Registers].
- **AcDcTimerxXX**: AC/DC wake alarm timer registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.17 [Wake Alarm Device (AcDcTimer) Registers].
- **AL2AHBxXX**: AL2AHB registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.18 [A-Link to AHB Bridge (AL2AHB) Configuration Registers].
- **DMAxXX**: DMA registers; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.19 [DMA Registers].
- **I2C[Y]xXX**: I^2C registers; Y specifies the controller number; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.20 [I2C Configuration Registers].
- **UART[Y]xXX**: UART registers; Y specifies the controller number; XX specifies the hexadecimal byte address offset. Unless otherwise specified, there is one set of these registers per node; the registers in a node are accessible to any core on that node. See 3.26.21 [UART Registers].

Each mnemonic may specify the location of one or more registers that share the same base definition. A mnemonic that specifies more than one register contains one or more ranges within braces. The ranges are specified as follows:
- Comma separated lists [A,B]: Define specific instances of a register (e.g., D0F3x[1,0]40 defines two registers D0F3x40 and D0F3x140).
- Colon separated ranges [A:B]: Defines all registers that contain the range between A and B. Examples:
 - D0F3x[50:40] defines five registers D0F3x40, D0F3x44, D0F3x48, D0F3x4C, and D0F3x50.
 - D[8:2]F0x40 defines seven registers D2F0x40, D3F0x40, D4F0x40, D5F0x40, D6F0x40, D7F0x40, and D8F0x40.
 - D0F0xE4_x013[2:0]_0000 defines three registers D0F0xE4_x0130_0000, D0F0xE4_x0131_0000, and D0F0xE4_x0132_0000.
- Colon separated ranges with a explicit step [A:BstepC]: Defines the registers from A to B, C defines the offset between registers (e.g., D0F3x[50:40:step8] defines three registers D0F3x40, D0F3x48, and D0F3x50).

The processor includes a single set of IO-space and configuration-space registers. However, APIC, CPUID, and MSR register spaces are implemented once per processor core. Access to IO-space and configuration space registers may require software-level techniques to ensure that no more than one core attempts to access a register at a time.

The following is terminology found in the register descriptions.

Table 75: Terminology in Register Descriptions

Term	Definition
BIOS	Software recommendation syntax. See 3.1.2 [Software Recommendation (BIOS, SBIOS)].
SBIOS	
See	Reference to remote definition.
Alias	The alias keyword allows the definition of a soft link between two registers. • X is an alias of Y: X is a soft link to the register Y. • X1, X2 are an alias of Y: Both X1 and X2 are soft links to Y.
IF	Allows conditional definition as a function of register fields. The syntax is: • IF (conditional-expression) THEN definition ENDIF. • IF (conditional-expression) THEN definition ELSE definition ENDIF. • IF (conditional-expression) THEN definition ELSEIF (conditional-expression) THEN definition ELSE definition ENDIF.
THEN	
ELSEIF	
ELSE	
ENDIF	
Access Types	
Read	Capable of being read by software.
Read-only	Capable of being read but not written by software.
Write	Capable of being written by software.
Write-only	Write-only. Capable of being written by software. Reads are undefined.
Read-write	Capable of being written by software and read by software.
Set-by-hardware	Register field is set high by hardware, set low by hardware, or updated by hardware.
Cleared-by-hardware	
Updated-by-hardware	
Updated-by-SMU	
Write-1-to-clear	Software must write a 1 to the bit in order to clear it. Writing a 0 to these bits has no affect.
Write-1-only	Software can set the bit high by writing a 1 to it. Writes of 0 have no effect.
Reset-applied	Takes effect on warm reset.
GP-read	GP exception occurs on read.
GP-write	GP exception occurs on write.
GP-read-write	GP exception occurs on a read or a write.
Per-core	One instance per core. Only valid for MMIO config space. Writes of these bits from one core only affect that core's register. Reads return the values appropriate to that core.
Per-compute-unit	One instance per compute unit. Writes of these bits from one core only affect that compute unit's register. Reads return the values appropriate to that compute unit. See 2.4.2.1 [Registers Shared by Cores in a Compute Unit].
SharedNC	All cores share the one instance per-compute-unit non-coherently; see 2.4.2.1 [Registers Shared by Cores in a Compute Unit]. Valid only with per-compute-unit.
Per-L2	One instance per L2 cache. See CPUID Fn8000_001D_EAX_x2[NumSharingCache].
Per-node	One instance per node. See 3.1.1 [Northbridge MSRs In Multi-Core Products].

Table 75: Terminology in Register Descriptions (Continued)

Term	Definition
Not-same-for-all	Provide indication as to whether all instances of a given register should be the same across all cores/nodes according to the following equation: SameOnAllCheckEnabled = (Writable && (same-for-all \| MSR) && ~(Not-same-for-all \|\| UpdatedByHw)). UpdatedByHw = (Updated-by-hardware \|\| Set-by-hardware \|\| Cleared-by-hardware \|\| Set-when-done \|\| Cleared-when-done).
Same-for-all	
Field Definitions	
Reserved	Field is reserved for future use. Software is required to preserve the state read from these bits when writing to the register. Software may not depend on the state of reserved fields nor on the ability of such fields to return the state previously written.
Unused	Field is reserved for future use. Software is not required to preserve the state read from these bits when writing to the register. Software may not depend on the state of unused fields nor on the ability of such fields to return the state previously written.
MBZ	Must be zero. If software attempts to set an MBZ bit to 1, a general-protection exception (#GP) occurs.
RAZ	Read as zero. Writes are ignored, unless RAZ is combined with Write-only or Write-1-only.
Reset Definitions	
Reset	The reset value of each register is provided below the mnemonic or in the field description. Unless otherwise noted, the register state matches the reset value when RESET_L is asserted (either a cold or a warm reset). Reset values may include: • **X**: an X in the reset value indicates that the field resets (warm or cold) to an unspecified state.
Cold reset	The field state is not affected by a warm reset (even if the field is labeled “cold reset: X”); it is placed into the reset state when PWROK is deasserted. See "Reset" above for the definition of characters that may be found in the cold reset value.
Value	The current value of a read-only field or register. A value statement explicitly defines the field or register as read-only and the value returned under all conditions including after reset events. A field labeled “Value:” does not have a separate reset definition.

3.1.1 Northbridge MSRs In Multi-Core Products

MSRs that control Northbridge functions are shared between all cores on the node in a multi-core processor (e.g., MSRC001_001F). If control of Northbridge functions is shared between software on all cores, software must ensure that only one core at a time is allowed to access the shared MSR. Some MSRs are conditionally shared; see D18F3x44[NbMcaToMstCpuEn].

3.1.2 Software Recommendation (BIOS, SBIOS)

The following keywords specify the recommended value to be set by software.

- BIOS: AMD BIOS.
- SBIOS: Platform BIOS.

Syntax: BIOS: integer-expression. Any of the supported tags can be substituted for BIOS.

If “BIOS:” occurs in a register field then the recommended value is applied to the field. If “BIOS:” occurs after

a register name but outside of a register field table row then the recommended value is applied to the width of the register.

3.1.3 See Keyword (See:)

There is a special meaning applied to the use of “See:” that differs from the use of See not followed by a “:”.

- See, not followed by a “:”, simply refers the reader to a document location that contains related information.
- See followed by a “:” is a shorthand notation that indicates that the definition for this register or register field inherits all properties and definitions from the register or register field that follows "See:". Any definition local to the register or register field supercedes this inheritance.

“See:” can be used in the following ways:

- Full register width. CPUID Fn0000_0001_EAX inherits it’s full register width definition from D18F3xFC.
- Register field. MSR0000_0277[PA1MemType] inherits it’s definition from PA0MemType, however, the local reset of 4h overrides the inherited PA0MemType reset of 6h.
- Valid values definition. MSR0000_020[E,C,A,8,6,4,2,0][MemType], for example, inherits the valid values definition from Table 235 [Valid Values for Memory Type Definition].

3.1.4 Mapping Tables

The following mapping table types are defined.

3.1.4.1 Register Mapping

The register mapping table specifies the specific function for each register in a range of registers.

Table 196, for example, specifies that the D18F5x160 function is for NB P-state 0.

3.1.4.2 Index Mapping

The index mapping table is similar to the register mapping table, but specifies the register by index instead of by full register mnemonic.

3.1.4.3 Field Mapping

The field mapping table maps the fields of a range of registers. The rows are the registers that are mapped. Each column specifies a field bit range that is mapped by that column for all registers. The cell at the intersection of the register and the field bit range specifies the suffix that is appended to the register field. “Reserved” specifies that the field is reserved for the register of that row.

3.1.4.4 Broadcast Mapping

The broadcast mapping table maps a register address to a range of register addresses that are written as a group when the broadcast register address is written. The register address is formed by the concatenation of the row address with the column address. The cell at the intersection of the row and column address is a range of register addresses that will be written as a group when the row and column address is written.

3.1.4.5 Reset Mapping

The reset mapping table specifies the reset, cold reset, or value for each register in a range of registers.

Table 230 [Reset Mapping for CPUID Fn8000_0000_E[D,C,B]X], for example, specifies that the CPUID Fn0000_0000_EBX register has a value of 6874_7541h, with a comment of "The ASCII characters "h t u A"".

3.1.4.6 Valid Values

The valid values table defines the valid values for one or more register fields. The valid values table is equivalent in function to the Bits/Description tables in register fields (e.g., MSR0000_0277[PA0MemType]) and is most often used when the table becomes too large and unwieldy to be included into the register field (e.g., Table 235 [Valid Values for Memory Type Definition]).

3.1.4.7 BIOS Recommendations

The BIOS recommendations table defines "BIOS:" recommendations that are conditional and complex enough to warrent a table.

Table 169 [BIOS Recommendations for D18F2x1B[4:0]], for example, specifies the BIOS recommendations for D18F2x1B0[DcqBwThrotWm] and D18F2x1B4[DcqBwThrotWm1, DcqBwThrotWm2]. All cells under the "Condition" header for a given row are ANDed to form the condition for the values to the right of the condition. For example, rows 1-3 and column 1 provide the following equivalent BIOS recommendation:

- D18F2x1B0[DcqBwThrotWm]: BIOS: IF (DdrRate == 667) THEN 4h. ELSEIF (DdrRate == 800) THEN 5h. ELSEIF (DdrRate == 1066) THEN 6h. ELSEIF etc. ENDIF.

3.2 IO Space Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

IOCF8 IO-Space Configuration Address

Reset: 0. IOCF8 [IO-Space Configuration Address], and IOCFC [IO-Space Configuration Data Port], are used to access system configuration space, as defined by the PCI specification. IOCF8 provides the address register and IOCFC provides the data port. Software sets up the configuration address by writing to IOCF8. Then, when an access is made to IOCFC, the processor generates the corresponding configuration access to the address specified in IOCF8. See 2.7 [Configuration Space].

IOCF8 may only be accessed through aligned, DW IO reads and writes; otherwise, the accesses are passed to the appropriate IO link. Accesses to IOCF8 and IOCFC received from an IO link are treated as all other IO transactions received from an IO link and are forwarded based on the settings in D18F1x[DC:C0] [IO-Space Base/Limit]. IOCF8 and IOCFC in the processor are not accessible from an IO link.

Bits	Description
31	**ConfigEn: configuration space enable**. Read-write. 1=IO read and write accesses to IOCFC are translated into configuration cycles at the configuration address specified by this register. 0=IO read and write accesses are passed to the appropriate IO link and no configuration access is generated.
30:28	Reserved.
27:24	**ExtRegNo: extended register number**. Read-write. ExtRegNo provides bits[11:8] and RegNo provides bits[7:2] of the byte address of the configuration register. ExtRegNo is reserved unless it is enabled by MSRC001_001F[EnableCf8ExtCfg].
23:16	**BusNo: bus number**. Read-write. Specifies the bus number of the configuration cycle.
15:11	**Device: bus number**. Read-write. Specifies the device number of the configuration cycle.
10:8	**Function**. Read-write. Specifies the function number of the configuration cycle.

7:2	**RegNo: register address**. Read-write. See IOCF8[ExtRegNo].
1:0	Reserved.

IOCFC IO-Space Configuration Data Port

Bits	Description
31:0	**Data**. Read-write. Reset: 0. See IOCF8.

3.3 Device 0 Function 0 (Root Complex) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D0F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 1576h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D0F0x04 Status/Command

Reset: 0000_0004h.

Bits	Description
31:21	Reserved.
20	**CapList: capability list**. Read-only. 1=Capability list supported.
19:3	Reserved.
2	**BusMasterEn: bus master enable**. Read-only.
1	**MemAccessEn: memory access enable**. Read-only.
0	**IoAccessEn: IO access enable**. Read-only.

D0F0x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode: class code**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D0F0x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:24	Reserved.
23	**DeviceType**. Read-only. 0=Single function device. 1=Multi-function device.
22:16	**HeaderType**. Read-only.
15:8	**LatencyTimer**. Read-only.
7:0	**CacheLineSize**. Read-only.

D0F0x2C Subsystem and Subvendor ID

Bits	Description
31:16	**SubsystemID**. Read-only. Value: 1410h.
15:0	**SubsystemVendorID**. Read-only. Value: 1022h.

D0F0x34 Capabilities Pointer

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. There is no capability list.

D0F0x48 NB Header Write Register

Reset: 0000_0080h.

Bits	Description
31:8	Reserved.
7	**DeviceType: device type**. Read-write. This field sets the value in the corresponding field in D0F0x0C[DeviceType]. 0=Single function device. 1=Multi-function device.
6:0	Reserved.

D0F0x4C PCI Control

Reset: 0000_0000h.

Bits	Description
31:27	Reserved.
26	**HPDis: hot plug message disable**. Read-write. 1=Hot plug message generation is disabled.
25:24	Reserved.
23	**MMIOEnable: memory mapped IO enable**. Read-write. 1=Decoding of MMIO cycles is enabled. The MMIO Base/Limit pair (D0F0x64_x17 and D0F0x64_x18) are decoded. This range is used to create an MMIO hole in the DRAM address range used for DMA decoding. DMA writes that fall into the MMIO range are treated as potential p2p requests. DMA reads that fall into the MMIO range are aborted as unsupported requests.
22:6	Reserved.
5	**SerrDis**: **system error message disable**. Read-write. 1=The generation of SERR messages is disabled.
4	**PMEDis: PME disable**. Read-write. 1=The generation of PME messages is disabled.
3:0	Reserved.

D0F0x60 Miscellaneous Index

Reset: 0000_0000h. The index/data pair registers, D0F0x60 and D0F0x64, are used to access the registers at D0F0x64_x[FF:00]. To access any of these registers, the address is first written into the index register, D0F0x60, and then the data is read from or written to the data register, D0F0x64.

Bits	Description
31:8	Reserved.
7:0	**MiscIndAddr: miscellaneous index register address**. Read-write.

D0F0x64 Miscellaneous Index Data

See D0F0x60. Address: D0F0x60[MiscIndAddr].

Bits	Description
31:0	**MiscIndData: miscellaneous index data register**.

D0F0x64_x00 Northbridge Control

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7	**HwInitWrLock**. Read-write. 1=Lock HWInit registers. 0=Unlock HWInit registers.
6:0	Reserved.

D0F0x64_x0C IOC Bridge Control

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D0F0x64_x0D IOC PCI Configuration

Bits	Description
31	**IommuDis**. Read-only. Value: Product specific.
30:1	Reserved.
0	**PciDev0Fn2RegEn**. Read-write. Reset: 1. 1=Enable configuration accesses to device 0 function 2.

D0F0x64_x16 IOC Advanced Error Reporting Control

Reset: 0000_0001h.

Bits	Description
31:1	Reserved.
0	**AerUrMsgEn: AER unsupported request message enable**. Read-write. 1=AER unsupported request messages are enabled.

D0F0x64_x17 Memory Mapped IO Base Address

Reset: 0000_0000h.

Bits	Description
31:0	**MmioBase[47:16]: memory mapped IO base address**. Read-write.

D0F0x64_x18 Memory Mapped IO Limit

Reset: 0000_0000h.

Bits	Description
31:0	**MmioLimit[47:16]: memory mapped IO limit**. Read-write.

D0F0x64_x19 Top of Memory 2 Low

Reset: 0000_0000h.

Bits	Description
31:23	**Tom2[31:23]: top of memory 2**. Read-write. BIOS: MSRC001_001D[TOM2[31:23]]. This field specifies the maximum system address for upstream read and write transactions that are forwarded to the host bridge. All addresses less than this system address are forwarded to DRAM and are not checked to determine if the transaction is a peer-to-peer transaction. All upstream reads with addresses greater than or equal to this system address are master aborted.
22:1	Reserved.
0	**TomEn: top of memory enable**. Read-write. BIOS: MSRC001_0010[MtrrTom2En]. 1=Top of memory check enabled.

D0F0x64_x1A Top of Memory 2 High

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**Tom2[39:32]: top of memory 2**. Read-write. BIOS: MSRC001_001D[TOM2[39:32]]. See D0F0x64_x19[Tom2].

D0F0x64_x1D Internal Graphics PCI Control

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3	**Vga16En: VGA IO 16 bit decoding enable**. Read-write. BIOS: D0F0x64_x1D[VgaEn]. 1=Address bits[15:10] for VGA IO cycles are decoded. 0=Address bits[15:10] for VGA IO cycles are ignored.
2	Reserved.
1	**VgaEn: VGA enable**. Read-write. 1=Enable VGA range in Intgfx.
0	Reserved.

D0F0x64_x1F FCH Location

Reset: 0004_0001h.

Bits	Description
31:16	**SBLocatedCore: Indicates which GPP Core has the FCH attached to it**. Read-write. Bits — Definition 0000h — No FCH attached. 0001h — FCH located under PCIE0. 0002h — FCH located under PCIE1. 0003h — Reserved. 0004h — FCH located under SBG. FFFFh-0005h — Reserved.
15:0	**SBLocatedPort: Indicates which Port on the SBLocatedCore has the FCH**. Read-write. Bits — Definition 0000h — No FCH attached. 0001h — FCH located on Port A of SBLocatedCore. 0002h — FCH located on Port B of SBLocatedCore. 0003h — Reserved. 0004h — FCH located on Port C of SBLocatedCore. 0007h-0005h — Reserved. 0008h — FCH located on Port D of SBLocatedCore. 000Fh-0009h — Reserved. 0010h — FCH located on Port Eof SBLocatedCore. FFFFh-0011h — Reserved.

D0F0x64_x22 LCLK Control 0

Reset: 7F3F_8100h.

Bits	Description
31	Reserved.
30	**SoftOverrideClk0**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for the host request path to the PCIe® cores.

29	**SoftOverrideClk1**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for the host request path to the internal graphics and the host response path.
28	**SoftOverrideClk2**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for the host configuration requests.
27	**SoftOverrideClk3**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for the debug bus path.
26	**SoftOverrideClk4**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for the host request path to the configuration block.
25	Reserved.
24	**SoftOverrideClk6**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for the VDM controller block.
23:0	Reserved.

D0F0x64_x23 LCLK Control 1

Reset: 7F3F_8100h.

Bits	Description
31	Reserved.
30	**SoftOverrideClk0**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for upstream DMA requests from all sources.
29	**SoftOverrideClk1**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for upstream DMA requests from the GPPFCH link core.
28	**SoftOverrideClk2**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for upstream DMA requests from internal graphics and its DMA response reordering path.
27	**SoftOverrideClk3**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for upstream DMA requests from internal graphics.
26	**SoftOverrideClk4**. Read-write. BIOS: 0. 1=Dynamic clock gating disabled for upstream DMA requests from the Gfx link core.
25:0	Reserved.

D0F0x64_x3[4:0] Programmable Device Remap Register

Table 76: Reset values for D0F0x64_x3[4:0]

Register	Reset	Function
D0F0x64_x30	0000_0011h	Program [7:3]DevNum, [2:0]FnNum to map to Port0 of PCIE0.
D0F0x64_x31	0000_0012h	Program [7:3]DevNum, [2:0]FnNum to map to Port1 of PCIE0.
D0F0x64_x32	0000_0013h	Program [7:3]DevNum, [2:0]FnNum to map to Port2 of PCIE0.
D0F0x64_x33	0000_0014h	Program [7:3]DevNum, [2:0]FnNum to map to Port3 of PCIE0.
D0F0x64_x34	0000_0015h	Program [7:3]DevNum, [2:0]FnNum to map to Port4 of PCIE0.

Software can only utilize device and function number combinations that are used by other (local) PCIe®

bridges. This effectively allows swapping of device and function numbers between bridges.

Bits	Description
31:8	Reserved.
7:0	**DevFnMap**. Read-write. Program [7:3]DevNum, [2:0]FnNum to map to PortA/B/C/D of each PCIe® core.

D0F0x64_x46 IOC Features Control

Reset: 0001_1023h.

Bits	Description
31:17	Reserved.
16	**CgttLclkOverride**. Read-write. BIOS: 0. Global bit to disable all LCLK gating branches.
15:3	Reserved.
2:1	**P2PMode: peer-to-peer mode**. Read-write. Specifies how upstream write transactions above D0F0x64_x19[Tom2] are completed. Bits — Definition 00b — Mode 0. Master abort writes that do not hit one of the internal PCI bridges. Forward writes that hit one of the internal PCI bridges to the bridge. 01b — Mode 1. Forward writes to the host bridge that do not hit one of the internal PCI bridges. Forward writes that hit one of the internal PCI bridges to the bridge. 10b — Mode 2. Forward all writes to the host bridge0. 11b — Reserved.
0	Reserved.

D0F0x64_x50 ACG Control

Reset: 0000_000Fh.

Bits	Description
31:8	Reserved.
7	**AcgPortBDis**. Read-write. 1=Disable bridge accesses to ACG Function 2.
6	**AcgPortBCfgDis**. Read-write. 1=Disable CFG accesses to ACG Function 2.
5	**AcgPortBP2PDis**. Read-write. 1=Disable P2P requests from ACG Function 2.
4	**AcgPortBBusMasterDis**. Read-write. 1=Disable DMA requests from ACG Function 2.
3:0	Reserved.

D0F0x7C IOC Configuration Control

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**ForceIntGfxDisable: internal graphics disable**. Read-write. Setting this bit disables the internal graphics and the HD Audio Controller.

D0F0x84 Link Arbitration

Bits	Description
31:10	Reserved.
9	**PmeTurnOff: PME_Turn_Off message trigger**. Read-write. Reset: 0. 1=Trigger a PME_Turn_Off message to all downstream devices if PmeMode == 1.
8	**PmeMode: PME message mode**. Read-write. Reset: 0. 1=PME_Turn_Off message is triggered by writing PmeTurnOff. 0=PME_Turn_Off message is triggered by a message from the FCH.
7:4	Reserved.
3	**VgaHole: vga memory hole**. Read-write. Reset: 1. This bit creates a hole in memory for the VGA memory range. 1=Requests hitting the VGA range are checked against PCI bridge memory ranges instead of being forwarded to system memory.
2:0	Reserved.

D0F0x90 Northbridge Top of Memory

Reset: 0000_0000h.

Bits	Description
31:23	**TopOfDram**. Read-write. BIOS: MSRC001_001A[TOM[31:23]]. Specifies the address that divides between MMIO and DRAM. From TopOfDram to 4G is MMIO; below TopOfDram is DRAM.
22:0	Reserved.

D0F0x94 Northbridge ORB Configuration Offset

Reset: 0000_0000h.
The index/data pair registers, D0F0x94 and D0F0x98, are used to access the registers at D0F0x98_x[FF:00]. To access any of these registers, the address is first written into the index register, D0F0x94, and then the data is read from or written to the data register, D0F0x98.

Bits	Description
31:7	Reserved.
6:0	**OrbIndAddr: ORB index register address**. Read-write.

D0F0x98 Northbridge ORB Configuration Data Port

See D0F0x94. Address: D0F0x94[OrbIndAddr].

Bits	Description
31:0	**OrbIndData: ORB index data register**.

D0F0x98_x02 ORB PGMEM Control

Reset: 0000_0000h.

Bits	Description
31:16	**PgmemHysteresis**. Read-write. Hysteresis value for power-gating of ORB TX group of memories. Value represents absolute number of LCLK cycles.
15:8	Reserved.
7:5	**TxPgmemStEn**. Read-write. Power-gating enablement for the ORB TX group of memories. Bits are mutually exclusive. Bit Definition [0] LS (light sleep) mode. [1] DS (deep sleep) mode. [2] SD (shutdown) mode. [3] Reserved.
4:2	**RxPgmemStEn**. Read-write. Power-gating enablement for the ORB RX group of memories. Bits are mutually exclusive. Bit Definition [0] LS (light sleep) mode. [1] DS (deep sleep) mode. [2] SD (shutdown) mode. [3] Reserved.
1	**OrbTxPgmemEn**. Read-write. Enables ORB TX memory power-gating.
0	**OrbRxPgmemEn**. Read-write. Enables ORB RX memory power-gating.

D0F0x98_x06 ORB Downstream Control 0

Reset: 0000_0000h.

Bits	Description
31:27	Reserved.
26	**UmiNpMemWrEn**. Read-write. BIOS: See 2.11.4. 1=NP protocol over UMI for memory-mapped writes targeting LPC enabled. This bit may be set to avoid a deadlock condition.
25:0	Reserved.

D0F0x98_x07 ORB Upstream Arbitration Control 0

Reset: 0000_0080h.

Bits	Description
31	**SMUCsrIsocEn**. Read-write. BIOS: 1. 1=CSR accesses go through ISOC channel. If this bit is set, D0F0x98_x1E[HiPriEn] must also be set.
30:17	Reserved.
16	**SyncFloodOnParityErr**. Read-write. Enable short circuit syncflood when arb_np detects a parity error for error containment.
15	**DropZeroMaskWrEn**. Read-write. BIOS: 1. 1=Drop byte write request that have all bytes masked. 0=Forward byte write request that have all bytes masked.
14:8	Reserved.
7	**IommuIsocPassPWMode**. Read-write. BIOS: 1. 1=Always set PassPW for IOMMU upstream isochronous requests.
6	**DmaReqRespPassPWMode**. Read-write. BIOS: 0. Specifies the RespPassPW bit for non-posted upstream DMA requests. Bits: Description 0b: Always 1. 1b: Value passed from IOC.
5	**SpgBwOptEn**. Read-write. BIOS: 1. 1=Optimize SPG byte write by detecting consecutive DW mask and translate the request to DW write.
4	**IommuBwOptEn**. Read-write. BIOS: 1. 1=Optimize IOMMU L2 byte write by detecting consecutive DW mask and translate the request to DW write.
3	**IocAtomicROMapDis**. Read-write. 1=Sets PassPW=0 for atomics. 0=Sets PassPW=RO for atomics.
2	**IocRdROMapDis**. Read-write. 1=Disable mapping relax ordering bit to RdRespPpw bit for IOC reads.
1	**IocWrROMapDis**. Read-write. 1=Disables mapping relax ordering bit to PassPW bit for IOC writes.
0	**IocBwOptEn**. Read-write. BIOS: 1. 1=Enable optimization of byte writes by detecting consecutive DW masks and translating the request to DW writes.

D0F0x98_x08 ORB Upstream Arbitration Control 1

This register specifies the weights of the weighted round-robin arbiter in stage 1 of the upstream arbitration for non-posted reads.

Bits	Description
31:24	**NpWrrLenD**. Read-write. Reset: 8h. BIOS: 8h. This field defines the maximum number of non-posted read requests from the SPG that are serviced before the arbiter switches to the next client.
23:16	**NpWrrLenC**. Read-write. Reset: 8h. BIOS: 1h. This field defines the maximum number of non-posted read requests from the SMU that are serviced before the arbiter switches to the next client.

15:8	**NpWrrLenB**. Read-write. Reset: 8h. BIOS: 8h. This field defines the maximum number of non-posted read requests from IOMMU that are serviced before the arbiter switches to the next client.
7:0	**NpWrrLenA**. Read-write. Reset: 8h. BIOS: 8h. This field defines the maximum number of non-posted read requests from IOC that are serviced before the arbiter switches to the next client.

D0F0x98_x09 ORB Upstream Arbitration Control 2

Reset: 0000_0808h.

This register specifies the weights of the weighted round-robin arbiter in stage 1 of the upstream arbitration for posted writes.

Bits	Description
31:24	Reserved.
23:16	**PWrrLenC**. Read-write. This field defines the maximum number of posted write requests from the SPG that are serviced before the arbiter switches to the next client.
15:8	**PWrrLenB**. Read-write. This field defines the maximum number of posted write requests from the IOMMUthat are serviced before the arbiter switches to the next client.
7:0	**PWrrLenA**. Read-write. This field defines the maximum number of posted write requests from the IOC that are serviced before the arbiter switches to the next client.

D0F0x98_x0C ORB Upstream Arbitration Control 5

Reset: 0000_0808h. This register specifies the weights of the weighted round-robin arbiter in stage 2 of the upstream arbitration.

Bits	Description
31:16	Reserved.
15:8	**GcmWrrLenB**. Read-write. BIOS: 08h. This field defines the maximum number of non-posted read requests from stage 1 that are getting serviced in the round-robin before the stage 2 arbiter switches to the next client.
7:0	**GcmWrrLenA**. Read-write. BIOS: 08h. This field defines the maximum number of posted write requests from stage 1 that are getting serviced in the round-robin before the stage 2 arbiter switches to the next client.

D0F0x98_x1E ORB Receive Control 0

Reset: 4800_0000h.

Bits	Description
31:24	**RxErrStatusDelay**. Read-write. BIOS: 48h. Delay error status by number of LCLK cycles to filter false errors caused by reset skew.
23:2	Reserved.

1	**HiPriEn**. Read-write. BIOS: 1. 1=High priority channel enabled. See D0F0x98_x27[IOMMUU-rAddr[31:6]]. IF (D0F0x98_x1E[HiPriEn] == 0) THEN (D0F0x98_x07[SMUCsrIsocEn] == 0). IF (D0F0x98_x1E[HiPriEn] == 1) THEN (D18F0x[E4,C4,A4,84][IsocEn] = 1) in order to fully enable the Isoc channel on the ONION Link.
0	Reserved.

D0F0x98_x26 ORB IOMMU Control 0

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**IOMMUUrAddr[39:32]**. Read-write. See: D0F0x98_x27[IOMMUUrAddr[31:6]].

D0F0x98_x27 ORB IOMMU Control 1

Reset: 0000_0000h.

Bits	Description
31:6	**IOMMUUrAddr[31:6]**. Read-write. BIOS: IOMMUUrAddr[39:6] must be programmed to a safe system memory address when D0F2x44[IommuEnable] == 1. IOMMUUrAddr[39:6] = {D0F0x98_x26[IOMMUUrAddr[39:32]], IOMMUUrAddr[31:6]}. IOMMU requests that are not directed to system memory are redirected to IOMMUUrAddr.
5:0	Reserved.

D0F0x98_x28 ORB Transmit Control 0

Reset: 0000_0002h.

Bits	Description
31:2	Reserved.
1	**ForceCoherentIntr**. Read-write. BIOS: 1. 1=Interrupt request are forced to have coherent bit set.
0	Reserved.

D0F0x98_x29 ORB Transmit Status

Reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**HostRdRespCrsDetected**. Read; Write-1-to-clear. 1=A host read response of type CRS was detected.

D0F0x98_x2C ORB Clock Control

Reset: 000F_0604h.

Bits	Description
31:16	**WakeHysteresis**.Read-write. BIOS: 19h. Specifies the amount of time hardware waits after ORB becomes idle before deasserting the wake signal to the NB. Wait time = WakeHysteresis * 200 ns. Changes to this field should be done prior to setting DynWakeEn.
15:11	Reserved.
10	**SBDmaActiveMaskIntFCH**. Read-write. BIOS: 1. 0=iFCH to ORB indicator triggers OnInbWake assertion. 1=iFCH to ORB indicator is masked out.
9	**SBDmaActiveMask**. Read-write. BIOS: 1. 0=SB_DMA_ACTIVE_L state affects OnInbWake state. 1=SB_DMA_ACTIVE_L state is masked out.
8:4	Reserved.
3	**FCHSel**. Read-write. 0=System is configured with dFCH. 1=System is configured with iFCH.
2	**CgttLclkOverride**. Read-write. BIOS: 0. Global bit to disable all LCLK gating branches in the ORB.
1	**DynWakeEn**.Read-write. BIOS: 1. 1=Enable dynamic toggling of the wake signal between ORB and NB. 0=Disable dynamic toggling of the wake signal. See WakeHysteresis.
0	Reserved.

D0F0x98_x37 ORB Allow LDTSTOP Control 0

Reset: 0020_0000h.

Bits	Description
31:28	Reserved.
27:16	**LDTStopHystersis**. Read-write. Specifies the number of timer periods (200 ns) the AllowLDTStop signal is held low before ORB asserts the signal again.
15:2	Reserved.
1	**DmaActiveOutEn**. Read-write. 1=Enable ORB to drive the DMAACTIVE_L pin. Meaningful only when D0F0x98_x37[AllowLDTStopPinMode] == 0.
0	**AllowLDTStopPinMode**. Read-write. Indicates the definition of the ALLOW_LDTSTOP pin. 0=Pin is used as DMAACTIVE_L. 1=Pin is used as ALLOW_LDTSTOP.

D0F0x98_x3A ORB Source Tag Translation Control 2

Reset: 0000_0000h.

Bits	Description
31:0	**ClumpingEn**. Read-write. BIOS should follow the below requirements. Valid only for PCIE0 and GBIF client clumping; internal unit ID ranges 4h-8h, 9h-Dh, and 14h-17h respectively. Legal PCIE0 clumping settings are: [8:4] == 0_0010b, applicable only in x0/0/0/0/8 system configuration. Legal GBIF clumping settings are: [23:20] == 0010b, 0110b and 1110b which are applicable in any system configuration. 1110b is the recommended value. All other bits of this register must always remain 0. See D18F0x[11C,118,114,110].

D0F0x98_x3B ORB Source Tag Translation Control 3

Reset: 0000_0000h.

Bits	Description
31:0	**IocOutstandingMask**. Read-write. Limit number of outstanding requests for every DMA client via the IOC.

D0F0x98_x4[A,9] ORB LCLK Clock Control 1-0

Reset: 7F3F_8100h.

Bits	Description
31	Reserved.
30	**SoftOverrideClk0**. Read-write. BIOS: 0. See SoftOverrideClk6.
29	**SoftOverrideClk1**. Read-write. BIOS: 0. See SoftOverrideClk6.
28	**SoftOverrideClk2**. Read-write. BIOS: 0. See SoftOverrideClk6.
27	**SoftOverrideClk3**. Read-write. BIOS: 0. See SoftOverrideClk6.
26	**SoftOverrideClk4**. Read-write. BIOS: 0. See SoftOverrideClk6.
25	**SoftOverrideClk5**. Read-write. BIOS: 0. See SoftOverrideClk6.

24	**SoftOverrideClk6**. Read-write. BIOS: 0. 1=Clock gating disabled. 0=Clock gating enabled.
23:0	Reserved.

D0F0xB8 SMU Index Address

The index/data pair registers, D0F0xB8 and D0F0xBC, are used to access the registers at D0F0xBC_x[FFFF_FFFF:0000_0000]. To access any of these registers, the address is first written into the index register, D0F0xB8, and then the data is read from or written to the data register, D0F0xBC.

Bits	Description
31:0	**NbSmuIndAddr: smu index address**. Read-write. Reset: 0.

D0F0xBC SMU Index Data

See D0F0xB8. Address: D0F0xB8[NbSmuIndAddr].

Bits	Description
31:0	**NbSmuIndData: smu index data**. Reset: 0.

D0F0xBC_x1300_007C MP1_C2PMSG_31

Bits	Description
31:0	**MP1_C2PMSG_31**. Read-write. Reset: 0. CPU to MP inbound message.

D0F0xBC_xC001_7150 SVI Loadline Configuration

Bits	Description
31:23	Reserved.
22:21	**SviLoadLineOffsetVddNb**. Read-only. Reset: value varies by product.
20:19	**SviLoadLineOffsetVdd**. Read-only. Reset: value varies by product.
18:16	Reserved.
15:13	**SviLoadLineTrimVddNb**. Read-only. Reset: value varies by product.
12:10	**SviLoadLineTrimVdd**. Read-only.Reset: value varies by product.
9:0	Reserved.

D0F0xBC_xD020_008C LCLK Deep Sleep Control

Bits	Description
31:3	Reserved.
2:0	**DIV_ID**. Read-write. Reset: 6. Bits: Description 000b: Clock OFF 001b: Divide by 1 010b: Divide by 2 011b: Divide by 4 100b: Divide by 8 101b: Divide by 16 110b: Divide by 32 111b: Divide by 1.5

D0F0xBC_xD021_1058 SPR P-State Power Information 1

Bits	Description
31:24	**PwrValue3: P3 power value**. Value: Product-specific. See PwrValue0.
23:16	**PwrValue2: P2 power value**. Value: Product-specific. See PwrValue0.
15:8	**PwrValue1: P1 power value**. Value: Product-specific. See PwrValue0.
7:0	**PwrValue0: P0 power value**. Value: Product-specific. PwrValue and PwrDiv together specify the expected power draw of a single core in P0 and 1/NumCores of the Northbridge in the NB P-state as specified by MSRC001_00[6B:64][NbPstate]. NumCores is defined to be the number of cores per node at cold reset. PwrDiv: Description 00b: PwrValue / 1 W, Range: 0 to 255 W 01b: PwrValue / 10 W, Range: 0 to 25.5 W 10b: PwrValue / 100 W, Range: 0 to 2.55 W 11b: Reserved

D0F0xBC_xD021_105C SPR P-State Power Information 2

Bits	Description
31:24	Reserved.
23:22	**PwrDiv7: P7 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
21:20	**PwrDiv6: P6 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
19:18	**PwrDiv5: P5 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
17:16	**PwrDiv4: P4 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
15:14	**PwrDiv3: P3 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
13:12	**PwrDiv2: P2 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
11:10	**PwrDiv1: P1 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].

9:8	**PwrDiv0: P0 power divisor**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
7:0	**PwrValue4: P4 power value**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].

D0F0xBC_xD021_1060 SPR P-State Power Information 3

Bits	Description
31:24	Reserved.
23:16	**PwrValue7: P7 power value**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
15:8	**PwrValue6: P6 power value**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].
7:0	**PwrValue5: P5 power value**. Value: Product-specific. See D0F0xBC_xD021_1058[PwrValue0].

D0F0xBC_xD021_1074 SPR Product Information Register 1

Bits	Description
31:21	Reserved.
20:17	**DiDtCfg4**. Value: Product-specific. See MSRC001_1028[DiDtCfg4].
16	**DiDtCfg3**. Value: Product-specific. See MSRC001_1028[DiDtCfg3]
15:14	**DiDtCfg2**. Value: Product-specific. See MSRC001_1028[DiDtCfg2].
13:6	**DiDtCfg1**. Value: Product-specific. See MSRC001_1028[DiDtCfg1].
5:1	**DiDtCfg0**. Value: Product-specific. See MSRC001_1028[DiDtCfg0].
0	**DiDtMode**. Value: Product-specific. See MSRC001_1028[DiDtMode].

D0F0xBC_xD021_1088 SPR Brand Name Address

D0F0xBC_xD021_1088 and D0F0xBC_xD021_108C provides the BIOS with a read-only name string that may be copied to MSRC001_00[35:30] at warm reset. Each of D0F0xBC_xD021_108C_x[B:0] is read as follows:

1. Write D0F0xBC_xD021_1088[Index].
2. Read D0F0xBC_xD021_108C[DATA].

Bits	Description
31:4	Reserved.
3:0	**Index: name string register index**. Read-write. Reset: 0. Bits: Description Bh-0h: Name String Register Index. Fh-Ch: Reserved.

D0F0xBC_xD021_108C SPR Brand Name Data Port

See D0F0xBC_xD021_1088 for register access information. Address: D0F0xBC_xD021_1088[Index].

Bits	Description
31:0	**DATA**. Read-only. Reset: 0.

D0F0xBC_xD021_108C_x[B:0] SPR Brand Name Data

Bits	Description
31:24	**NameStringByte3: name string ASCII character 3**. Read-only. Value: Product-specific.
23:16	**NameStringByte2: name string ASCII character 2**. Read-only. Value: Product-specific.
15:8	**NameStringByte1: name string ASCII character 1**. Read-only. Value: Product-specific.
7:0	**NameStringByte0: name string ASCII character 0**. Read-only. Value: Product-specific.

D0F0xBC_xD820_0C64 Hardware Temperature Control (HTC)

See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)].

Bits	Description
31	Reserved.
30:28	**HTC_PSTATE_LIMIT: HTC P-state limit select**. Read-only; Updated-by-SMU. Reset: Product-specific. Specifies the P-state limit of all cores when in the HTC-active state.
27:24	**HTC_HYST_LMT: HTC hysteresis limit**. Read-only; Updated-by-SMU. Reset: Product-specific. The processor exits the HTC-active state when (T_{ctl} < (HTC_TMP_LMT - HTC_HYST_LMT). Bits: Description 0h: 0 1h: 0.5 Eh-2h: <HTC_HYST_LMT*0.5> Fh: 7.5
23	**HTC_SLEW_SEL: HTC slew-controlled temperature select**. Read-only; Updated-by-SMU. Reset: 0. 1=HTC logic is driven by the slew-controlled temperature, T_{ctl}, specified in D0F0xBC_xD820_0CA4. 0=HTC logic is driven by the measured control temperature with no slew controls.
22:16	**HTC_TMP_LMT: HTC temperature limit**. Read-only; Updated-by-SMU. Reset: Product-specific. The processor enters the HTC-active state when T_{ctl} reaches or exceeds the temperature limit defined by this register. Bits: Description 00h: 52 01h: 52.5 7Eh-02h: <(HTC_TMP_LMT*0.5) + 52> 7Fh: 115.5
15:12	Reserved.
11	**ProcHotToGnbEn**. Read-only; Updated-by-SMU. Reset: 0. 1=The GPU is placed into a low-power state when PROCHOT_L is asserted. Note: the GPU power-state transitions associated with PROCHOT_L nominally occur every 1 millisecond; PROCHOT_L assertions and deassertions for less than this period may not result in GPU state changes.
10	**HtcToGnbEn**. Read-only; Updated-by-SMU. Reset: 0. 1=The GPU is placed into a low-power state when an HTC event occurs.
9:8	Reserved.

7	**HTC_APIC_LO_EN: P-state limit lower value change APIC interrupt enable**. Read-only; Updated-by-SMU. Reset: 0. HTC_APIC_LO_EN and HTC_APIC_HI_EN enable interrupts using APIC330 [LVT Thermal Sensor] of each core when the active P-state limit in MSRC001_0061[CurP-stateLimit] changes. HTC_APIC_LO_EN enables the interrupt when the limit value becomes lower (indicating higher performance). HTC_APIC_HI_EN enables the interrupt when the limit value becomes higher (indicating lower performance). 1=Enable interrupt.
6	**HTC_APIC_HI_EN: P-state limit higher value change APIC interrupt enable**. Read-only; Updated-by-SMU. Reset: 0. See HTC_APIC_LO_EN.
5	**HTC_ACTIVE_LOG: HTC-active log**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. This bit is set by hardware when the processor enters the HTC-active state. It is cleared by writing a 1 to it.
4	**HTC_ACTIVE: HTC-active state**. Read-only; Updated-by-hardware. Reset: X. 1=The processor is currently in the HTC-active state. 0=The processor is not in the HTC-active state.
3:1	Reserved.
0	**HTC_EN: HTC enable**. Read-only; Updated-by-SMU. Reset: 0. 1=HTC is enabled; the processor is capable of entering the HTC-active state.

D0F0xBC_xD820_0CA4 Reported Temperature Control

The slew rate controls in this register are used to filter processor temperature measurements. Separate controls are provided for a measured temperature that is higher or lower than T_{ctl}. The per-step timer counts as long as the measured temperature stays either above or below T_{ctl}. Each time the measured temperature changes to the other side of T_{ctl}, the step timer resets, and T_{ctl} is not changed. If, for example, step times are enabled in both directions, T_{ctl}=62.625, and the measured temperature keeps jumping quickly between 62.5 and 63.0, then (assuming the step times are long enough) T_{ctl} would not change. However, once the measured temperature settles on one side of T_{ctl}, T_{ctl} can step toward the measured temperature. If the difference of measured temperature minus T_{ctl} is greater than the value set by TMP_MAX_DIFF_UP, then T_{ctl} is set equal to the measured temperature. See 2.10 [Thermal Functions].

Bits	Description
31:21	**CUR_TEMP: current temperature**. IF (D0F0xBC_xD820_0CA4[CUR_TEMP_TJ_SEL] == 11b) THEN Read-write. ELSE Read-only; Updated-by-hardware. ENDIF. Reset: X. Provides the current control temperature, T_{ctl}, after the slew-rate controls have been applied. RangeUnajusted = ((D0F0xBC_xD820_0CA4[CUR_TEMP_TJ_SEL] != 11b). Bits / RangeUnajusted / Description 000h / 0 / -49 001h / 0 / -48.875 7FEh-002h / 0 / <(CurTmp*0.125)-49> 7FFh / 0 / 206.875 000h / 1 / 0 001h / 1 / 0.125 7FEh-002h / 1 / <CurTmp*0.125> 7FFh / 1 / 255.875
20	Reserved.
19:18	Reserved.

17:16	**CUR_TEMP_TJ_SEL: current temperature select**. Read-write. Reset: 00b. These bits may be used for diagnostic software. Bits Description 00b CurTmp provides the read-only T_{ctl} value. 01b Reserved. 10b Reserved. 11b CurTmp is a read-write register that specifies a value used to create T_{ctl}. The two LSBs are read-only zero.
15:13	Reserved.
12:8	**PER_STEP_TIME_DN: per step time down**. Read-write. Cold reset: 18h. BIOS: 0Fh. Specifies the time that measured temperature must remain below T_{ctl} before applying a 0.125 downward step. See TMP_MAX_DIFF_UP.
7	**TMP_SLEW_DN_EN: temperature slew downward enable**. Read-write. Cold reset: 0. BIOS: 1. 1=Downward slewing enabled. 0=Downward slewing disabled.
6:5	**TMP_MAX_DIFF_UP: temperature maximum difference up**. Read-write. Cold reset: 00b. BIOS: 11b. Specifies the maximum difference, (measured temperature - T_{ctl}), when T_{ctl} immediately updates to the measured temperature. Bits Description 00b 0.0 (disable upward slew) 01b 1.0 10b 3.0 11b 9.0
4:0	**PER_STEP_TIME_UP: per 1/8th degree step time up**. Read-write. Cold reset: 00h. BIOS: 0Fh. Specifies the time that measured temperature must remain above T_{ctl} before applying a 0.125 upward step. Bits Definition 1Fh-00h <(PER_STEP_TIME_UP[2:0] + 1) * 10^PER_STEP_TIME_UP[4:3]> ms, ranging from 1 ms to 8000 ms.

D0F0xBC_xD820_0CE4 Thermtrip Status

Bits	Description
31	**SW_THERM_TP: software THERMTRIP**. Write-1-only; Cleared-by-hardware. Reset: 0. Writing a 1 to this bit position induces a THERMTRIP event. This bit returns 0 when read. This is a diagnostic bit, and it should be used for testing purposes only.
30:6	Reserved.
5	**THERM_TP_EN: THERMTRIP enable**.Read-only. Reset: Product-specific. 1=The THERMTRIP state is supported. See 2.10.3.3 [THERMTRIP].
4	Reserved.
3	**THERM_TP_SENSE: THERMTRIP sense**. Read-only. Cold reset: 0. 1=The processor temperature exceeded the THERMTRIP value (regardless as to whether the THERMTRIP state is enabled). This bit is also set when the diagnostic bit SW_THERM_TP = 1.
2	Reserved.

1	**THERM_TP: THERMTRIP**. Read-only. Cold reset: 0. 1=The processor has entered the THERM-TRIP state.
0	Reserved.

D0F0xBC_xD822_20B8 COR0_CLK PLL LOCK TIMER

Bits	Description
31:29	Reserved.
28:16	**Pll_LONG_LOCK_TIMER**. Read-write. Reset: 0190h. This field alters the timing from PLL lock enable to Pll frequency lock, if BandGap is powered off.
15:13	Reserved.
12:0	**Pll_LOCK_TIMER**. Read-write. Reset: 00C8h. This field alters the timing from PLL lock enable to Pll frequency lock.

D0F0xBC_xD822_2114 COR0_CLK Gater Sequence Register

Bits	Description
31:16	Reserved.
15:8	**GatersOffTime**. Read-write. Reset: 03h. Represents number of REFCLK cycles between each gater disable.
7:0	**GatersOnTime**. Read-write. Reset: 03h. Represents number of REFCLK cycles between each gater enable.

D0F0xBC_xD823_0F00 SMUSVI Defaults

Bits	Description
31:2	Reserved.
1	**COF_VID_PROG**. Read-write. Reset: Product-specific.
0	Reserved.

D0F0xC8 DEV Index Address

The index/data pair registers, D0F0xC8 and D0F0xCC are used to access the registers at D0F0xCC_x[FF:00]. To access any of these registers, the address is first written into the index register, D0F0xC8, and then the data is read from or written to the data register, D0F0xCC. Specific IOC bridges (Device/Function) are selected using the D0F0xC8[NbDevIndSel] field and enumerated as *_ib[23:19,15:11]* in the indexed register's mnemonic.

Bits	Description
31:24	Reserved.

23:16	**NbDevIndSel: Device selector**. Read-write. Reset: 0. Bits Definition Bits Definition 10h-00h Reserved 19h D3F1 11h D2F1 1Ah D3F2 12h D2F2 1Bh D3F3 13h D2F3 1Ch D3F4 14h D2F4 1Dh D3F5 15h D2F5 FFh-1Eh Reserved 18h-16h Reserved
15:7	Reserved.
6:0	**NbDevIndAddr: Bridge (Device) index address**. Read-write. Reset: 0.

D0F0xCC DEV Index Data

See D0F0xC8. Address: D0F0xC8[NbDevIndAddr].

Bits	Description
31:0	**NbDevIndData: dev index data**.

D0F0xCC_x01_ib[23:19,15:11] IOC Bridge Control

Reset: 0000_0000h.

Bits	Description
31:24	**ApicRange**. Read-write. Sets the bridge APIC range.
23	**ApicEnable**. Read-write. 1=Enables the bridge APIC range decoding. Requests fall in bridge APIC range if addr[39:12]={00_FECh, APIC_Range[7:0]}.
22:21	Reserved.
20	**SetPowEn**. Read-write. 1=Enable generation of set_slot_power message to the bridge.
19	Reserved.
18	**CrsEnable**. Read-write. 1=Enables the hardware retry on receiving configuration request retry status.
17	**ExtDevCrsEn**. Read-write. 1=Reset the bridge CRS counter when an external device is plugged in or the link is down.
16	**ExtDevPlug**. Read-write. 1=Indicates to IOC that an external device is being plugged on the bridge.
15:7	Reserved.
6	**MaskUREn**. Read-write. 1=Convert UR status to SUCCESS status. Applies only to host completions.
5	**VdmDis**. Read-write. 1=Disables VDM accesses to this bridge.
4	Reserved.
3	**P2pDis**. Read-write. 1=Disables local peer-to-peer transactions forwarded to this bridge.
2	**CfgDis**. Read-write. 1=Configuration accesses to this bridge are disabled. Non-FCH bridges are not expected to set this bit.

1	**BusMasterDis**. Read-write. 1=The bridge's ability to operate as a bus master is disabled. This overrides the Bus Master Enable bit in the bridge.
0	**BridgeDis**. Read-write. BIOS: 1. 1=The bridge is hidden and no accesses are allowed to this bridge. When set to 1, transactions are routed to the FCH.

D0F0xCC_x02_ib[23:19,15:11] IOC Bridge Status

Reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**MaskURStatus**. Read; Write-1-to-clear. 1=A host completion with UR status was masked.

D0F0xD0 GBIF Index Address

The index/data pair registers, D0F0xD0 and D0F0xD4 are used to access the registers at D0F0xD4_x[FFFF_FFFF:0000_0000]. To access any of these registers, the address is first written into the index register, D0F0xD0, and then the data is read from or written to the data register, D0F0xD4.

Bits	Description
31:0	**NbGbifIndAddr: Gbif index address**. Read-write. Reset: 0.

D0F0xD4 GBIF Index Data

See D0F0xD0. Address: D0F0xD0[NbGbifIndAddr].

Bits	Description
31:0	**NbGbifIndData: Gbif index data**. Reset: 0.

D0F0xD4_x0109_14E1 CC Bif Bx Strap0 Ind

Reset: 0000_C004h.

Bits	Description
31:13	Reserved.
12	**StrapBifDoorbellBarDis**. Read-write.
11:6	Reserved.
5:3	**StrapBifMemApSize**. Read-write. Size of the primary memory apertures claimed in the PCI configuration space. The aperture size should be rounded to the next power of 2 size (up to 128MB) as the frame buffer for frame buffers smaller than 128MB. For frame buffers larger than 128 MB the aperture size should be set to 256 MB. See D0F0xD4_x0109_1507[StrapBifMemApSizePin]. Bits Definition 000b 128 MB 001b 256 MB 010b 64 MB 111b-011b Reserved

2:1	**StrapBifRegApSize**. Read-write. Specifies the register aperture size. Bits Definition 00b 64 MB 01b 128 MB 10b 256 MB 11b 512 MB
0	Reserved.

D0F0xD4_x0109_14E2 CC Bif Bx Strap1 Ind

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3	**StrapBifF064BarDisA**. Read-write.
2	Reserved.
1	**StrapBifIoBarDis**. Read-write.
0	Reserved.

D0F0xD4_x0109_1507 CC Bif Bx Pinstrap0 Ind

Reset: 0000_0802h.

Bits	Description
31:8	Reserved.
7:5	**StrapBifMemApSizePin**. Read-write. See: D0F0xD4_x0109_14E1[StrapBifMemApSize].
4:0	Reserved.

D0F0xE0 Link Index Address

Reset: 0130_8001h.

D0F0xE0 and D0F0xE4 are used to access D0F0xE4_x[FFFF_FFFF:0000_0000]. To read or write to one of these register, the address is written first into the address register D0F0xE0 and then the data is read from or written to the data register D0F0xE4.

The phy index registers (D0F0xE4_x0[2:1]XX_XXXX]) mapping to a specific phy, pin or pin group is shown in a table in the register definition. For example, to perform a read or write operation to configure Gfx phy 0 (P_GFX_[T,R]X[P,N][7:0] pin group) compensation, software should program D0F0xE0[31:0] = 0120_0000h. Accessing any register number that is not listed in the mapping table may result in undefined behavior.

Some phy registers support broadcast write operations to groups of 4 or 8 lanes. For example, to perform broadcast write operation to configure Gfx Link[3:0] (P_GFX_RX[P,N][3:0] lanes) receiver phase loop filter, software should program D0F0xE0[31:0] = 0120_5602h.

Bits	Description
31:24	**BlockSelect: block select**. Read-write. This field is used to select the specific register block to access. The encodings supported depends on the FrameType selected. BlockSelect must be set to 1.
23:16	**FrameType: frame type**. Read-write. This field is used to select the type of register block to access. Bits — Destination 1Nh — Phy interface block registers. 2Nh — Phy registers. 3Nh — Wrapper RefClk registers. 4Nh — Core registers. 5Nh — Wrapper TxClk registers. N — Register Block 0h — GPP/UMI links 1h — Gfx PCIe® links 3h-Fh — Reserved
15:0	**PcieIndxAddr: index address**. Read-write.

D0F0xE4 Link Index Data

See D0F0xE0. Address: {D0F0xE0[BlockSelect],D0F0xE0[FrameType],D0F0xE0[PcieIndxAddr]}.

Bits	Description
31:0	**PcieIndxData: index data**.

3.3.1 PIF Registers

Table 77: Mapping for PIF registers

D0F0xE0[31:16]	Wrapper	Port Description
0110h	PSD0	GPP+FCH
0111h	PSD1	Gfx

3.3.2 Wrapper Registers

Table 78: Mapping for wrapper registers

D0F0xE0[31:16]	Wrapper	Port Description
0130h	PSD	GPP/UMI
0131h	PPD	Gfx

D0F0xE4_x0150_0046 Subsystem and Vendor ID

Bits	Description
31:16	**SubsystemID: subystem id**. Read-write. Reset: 1234h. Specifies the value returned by D2F[5:1]xC4[SubsystemID].
15:0	**SubsystemVendorID: subsystem vendor id**. Read-write. Reset: 1022h. Specifies the value returned by D2F[5:1]xC4[SubsystemVendorID].

D0F0xE4_x0150_0080 Link Configuration

Reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3:0	**StrapBifLinkConfig**. Read-write; strap. Reset: Product-specific. BIOS: See Table 54. Bits Definition Bits Definition 0000b x16 IO Link (Gfx Only) 0100b 4 x1 IO Links (GPPFCH Only) 0001b x4 IO Link (GPPFCH Only) 0101b 2 x8 IO Links (Gfx Only) 0010b 2 x2 IO Links (GPPFCH Only) 011Xb Reserved 0011b 1 x2 IO Link, 2 x1 IO Links (GPPFCH Only) 1XXXb Reserved

D0F0xE4_x0150_0[C:8]00 Link Hold Training Control

Bits	Description
31:1	Reserved.
0	**HoldTraining: hold link training**. Read-write. Reset: 1. 1=Hold training on link.

D0F0xE4_x0150_0[C:8]03 Link Deemphasis Control

Bits	Description
31:6	Reserved.
5	**StrapBifDeemphasisSel**. Read-write; strap. Reset: 1. Controls the default value of D2F[5:1]x88[SelectableDeemphasis]. 1=RC advertises -3.5dB. 0=RC advertises -6dB.
4:0	Reserved.

D0F0xE4_x0130_8002 IO Link Wrapper Scratch

Cold reset: 0000_0000h.

Bits	Description
31:0	**PcieWrapScratch: Scratch**. Read-write.

D0F0xE4_x0130_80F0 BIOS Timer

Reset: 0000_0000h.

Bits	Description
31:0	**MicroSeconds**. Read-write; Updated-by-hardware. This field increments once every microsecond when the timer is enabled. The counter rolls over and continues counting when it reaches its FFFF_FFFFh. A write to this register causes the counter to reset and begin counting from the value written.

D0F0xE4_x0130_80F1 BIOS Timer Control

Reset: 0000_0064h.

Bits	Description
31:8	Reserved.
7:0	**ClockRate**. Read-write. Specifies the frequency of the reference clock in 1 MHz increments. Bits Definition 00h Timer disabled FFh-01h <ClockRate> MHz

3.3.3 IO Link Registers

D0F0xE4_x0140_0002 IO Link Hardware Debug

Reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**HwDebug[0]: ignore DLLPs in L1**. Read-write. BIOS: 1. 1=DLLPs are ignored in L1 so the TXCLK can be turned off.

D0F0xE4_x0140_0010 IO Link Control 1

Reset: 80E3_110Bh.

Bits	Description
31:13	Reserved.

12:10	**RxUmiAdjPayloadSize**. Read-write. BIOS: 100b. Payload size for the UMI link. Bits Definition Bits Definition 00Xb Reserved. 100b 64 bytes. 010b 16 bytes. 101b Reserved. 011b 32 bytes. 11Xb Reserved.
9	**UmiNpMemWrite: memory write mapping enable**. Read-write. 1=Internal non-posted memory writes are transferred to UMI.
8:4	Reserved.
3:1	**LcHotPlugDelSel: enhanced hot plug counter select**. Read-write. Bits Definition Bits Definition 0h 15 ms 4h 150 ms 1h 20 ms 5h 200 ms 2h 50 ms 6h 275 ms 3h 100 ms 7h 335 ms
0	**HwInitWrLock: hardware init write lock**. Read-write. 1=Lock HWInit registers. 0=Unlock HWInit registers.

D0F0xE4_x0140_0011 IO Link Config Control

Reset: 0000_000Fh.

Bits	Description
31:4	Reserved.
3:0	**DynClkLatency: dynamic clock latency**. Read-write. BIOS: See 2.11.4.3.1 [Link Configuration and Core Initialization]. Specifies the number of clock cycles after logic goes idle before clocks are gated off.

D0F0xE4_x0140_001C IO Link Control 2

Reset: 0E00_0109h.

Bits	Description
31:11	Reserved.
10:6	**TxArbMstLimit: transmitter arbitration master limit**. Read-write. BIOS: 4h. Defines together with TxArbSlvLimit a round-robin arbitration pattern for downstream accesses. TxArbMstLimit defines the weight for downstream CPU requests and TxArbSlvLimit for the downstream read responses.
5:1	**TxArbSlvLimit: transmitter arbitration slave limit**. Read-write. BIOS: 4h. See TxArbMstLimit for details
0	**TxArbRoundRobinEn: transmitter round-robin arbitration enabled**. Read-write. BIOS: 1. 1=Enable transmitter round-robin arbitration. 0=Disable transmitter round-robin arbitration.

D0F0xE4_x0140_0020 IO Link Chip Interface Control

Reset: 0000_0050h.

Bits	Description
31:10	Reserved.
9	**CiRcOrderingDis: chip interface RC ordering disable**. Read-write. 0=RC ordering logic is enabled. 1=RC ordering logic is disabled.
8:0	Reserved.

D0F0xE4_x0140_0040 IO Link Phy Control

Reset: 0001_0000h.

Bits	Description
31:16	Reserved.
15:14	**PElecIdleMode: electrical idle mode for physical layer**. Read-write. BIOS: 01b. Defines which electrical idle signal is used, either inferred by link controller of from phy. Bits — Definition 00b — Gen1 - entry:PHY, exit:PHY; Gen2 - entry:INF, exit:PHY. 01b — Gen1 - entry:INF, exit:PHY; Gen2 - entry:INF, exit:PHY. 10b — Gen1 - entry:PHY, exit:PHY; Gen2 - entry:PHY, exit:PHY. 11b — Gen1 - entry: PHY, exit: PHY; Gen2 - entry: PHY, exit: PHY.
13:0	Reserved.

D0F0xE4_x0140_00B0 IO Link Strap Control

Reset: 0000_8001h.

Bits	Description
31:6	Reserved.
5	**StrapF0AerEn**. Read-write. BIOS: 1. 1=AER support enabled. 0=AER support disabled.
4:3	Reserved.
2	**StrapF0MsiEn**. Read-write. BIOS: 1. Overrides MSI enable.
1:0	Reserved.

D0F0xE4_x0140_00C0 IO Link Strap Miscellaneous

Bits	Description
31	Reserved.
30	**StrapFlrEn**. Read-write.
29	**StrapMstAdr64En**. Read-write.
28	**StrapReverseAll**. Read-write. Reset: 0.
27:0	Reserved.

D0F0xE4_x0140_00C1 IO Link Strap Miscellaneous2

Bits	Description
31:4	Reserved.
3	**StrapGen3Compliance**. Read-write.
2	Reserved.
1	**StrapGen2Compliance**. Read-write. Reset: 1.
0	**StrapLinkBwNotificationCapEn**. Read-write. Reset: 0.

D0F0xF8 Northbridge IOAPIC Index

Reset: 0000_0000h. The index/data pair registers, D0F0xF8 and D0F0xFC, are used to access the registers at D0F0xFC_x[FF:00]. To access any of these registers, the address is first written into the index register, D0F0xF8, and then the data is read from or written to the data register, D0F0xFC.

Bits	Description
31:8	Reserved.
7:0	**IOAPICIndAddr: IOAPIC index register address**. Read-write.

D0F0xFC Northbridge IOAPIC Data

Reset: 0000_0000h. See D0F0xF8. Address: D0F0xF8[IOAPICIndAddr].

Bits	Description
31:0	**IOAPICIndData: IOAPIC index data register**. Read-write.

D0F0xFC_x00 IOAPIC Feature Control Register

Reset: 0000_0004h.

Bits	Description
31:5	Reserved.
4	**IoapicSbFeatureEn**. Read-write. 1=Enable masked interrupts to be routed back to the FCH PIC/IOAPIC.
3	Reserved.
2	**IoapicIdExtEn**. Read-write. Extend the IOAPIC ID from 4-bit to 8-bit. 0=4-bit ID. 1=8-bit ID.
1	Reserved.
0	**IoapicEnable**. Read-write. BIOS: 1. 1=Enables the INTGEN block to decode IOAPIC addresses. BIOS should always set this bit after programming the IOAPIC BAR in the init sequence.

D0F0xFC_x01 IOAPIC Base Address Lower

Reset: FEC0_0000h. See 3.16 [Northbridge IOAPIC Registers].

Bits	Description
31:8	**IoapicAddr**. Read-write. IOAPIC Base Address bits[31:8].
7:0	Reserved.

D0F0xFC_x02 IOAPIC Base Address Upper

Reset: 0000_0000h. See 3.16 [Northbridge IOAPIC Registers].

Bits	Description
31:0	**IoapicAddrUpper**. Read-write. IOAPIC Base Address bits[63:32].

D0F0xFC_x0F IOAPIC GBIF Interrupt Routing Register

Reset: 0000_0000h.

Bits	Description
31:6	Reserved.
5:4	**GBIFExtIntrSwz**. Read-write. Swizzle GBIF INTA/B/C/D based on the value in this field before mapping them onto the IOAPIC pins. Bits — Interrupt Swizzling 00b — ABCD 01b — BCDA 10b — CDAB 11b — DABC
3	Reserved.
2:0	**GBIFExtIntrGrp**. Read-write. Map GBIF INTA/B/C/D to IOAPIC pins [((grp+1)*4)-1:(grp*4)]. For GBIF, only INTA/B are used. INTC/D should be tied off.

D0F0xFC_x1[4:0] IOAPIC BR Interrupt Routing Register

Reset: 0000_0000h.

Bits	Description
31:21	Reserved.
20:16	**BrIntIntrMap**. Read-write. Map bridge n interrupts to IOAPIC redirection table entry.
15:6	Reserved.
5:4	**BrExtIntrSwz**. Read-write. Swizzle bridge n external INTA/B/C/D based on the value in this field before mapping them onto the IOAPIC pins. Bits — Definition 00b — ABCD 01b — BCDA 10b — CDAB 11b — DABC

3	Reserved.
2:0	**BrExtIntrGrp**. Read-write. Map bridge n external INTA/B/C/D to IOAPIC pins [((grp+1)*4)-1:(grp*4)].

D0F0xFC_x2F IOAPIC APG Interrupt Routing Register

Reset: 0000_0000h.

Bits	Description
31:6	Reserved.
5:4	**ApgPortAExtIntrSwz**. Read-write. Swizzle ACG PortA/ACG PortB INT A/B/C/D before maping to IOAPIC table/pins. Bits Definition Bits Definition 0h ABCD 2h CDAB 1h BCDA 3h DABC
3	Reserved.
2:0	**ApgPortAExtIntrGrp**. Read-write. Map ACG PortA/ACG PortB INT A/B/C/D to IOAPIC pins [((grp+1)*4)-1:(grp*4)].

D0F0xFC_x30 IOAPIC SPG Interrupt Routing Register

Reset: 0000_0000h.

Bits	Description
31:6	Reserved.
5:4	**SpgExtIntrSwz**. Read-write. Swizzle SPG INTA/B/C/D before mapping to IOAPIC pins. Bits Definition 0h ABCD 1h BCDA 2h CDAB 3h DABC
3	Reserved.
2:0	**SpgExtIntrGrp**. Read-write. Map SPG INTA/B/C/D to IOAPIC pins [((grp+1)*4)-1 : (grp*4)].

D0F0xFC_x31 IOAPIC Serial IRQ Status

Reset: 0000_0000h.

Bits	Description
31:0	**InternalIrqSts**. Read-only. Shows the status of the 32 IOAPIC interrupt pins.

D0F0xFC_x3[F:E] IOAPIC Scratch [1:0] Register

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch**. Read-write.

3.4 Device 0 Function 2 (IOMMU) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space]. See 2.12.1 [IOMMU Configuration Space].

D0F2x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 1577h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D0F2x04 Status/Command

Bits	Description
31	**ParityErrorDetected**. Read; Write-1-to-clear. Reset: 0.
30	**SignaledSystemError**. Read-only. Reset: 0.
29	**ReceivedMasterAbort**. Read; Write-1-to-clear. Reset: 0.
28	**ReceivedTargetAbort**. Read; Write-1-to-clear. Reset: 0.
27	**SignalTargetAbort**. Read-only. Reset: 0.
26:25	Reserved.
24	**MasterDataError**. Read; Write-1-to-clear. Reset: 0.
23:21	Reserved.
20	**CapList**. Read-only. Reset: 1. 1=Capability list supported.
19	**IntStatus**. Read-only. Reset: 0. 1=INTx message pending.
18:11	Reserved.
10	**InterruptDis**. Read-write. Reset: 0. 1=INTx interrupt message generation disabled.
9	Reserved.
8	**SerrEn**. Read-only. Reset: 0. 1=Enables reporting of non-fatal and fatal errors detected.
7	Reserved.
6	**ParityErrorEn**. Read-write. Reset: 0. 1=Enables setting of ParityErrorDetected status bit.
5:3	Reserved.
2	**BusMasterEn**. Read-write. Reset: 0. 1=Enables DMA request generation.
1	**MemAccessEn**. Read-only. Reset: 0.
0	**IoAccessEn**. Read-only. Reset: 0.

D0F2x08 Class Code/Revision ID

Reset: 0806_00XXh.

Bits	Description
31:8	**ClassCode: class code**. Read-only. Provides the IOMMU class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D0F2x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:24	**BIST**. Read-only.
23:16	**HeaderTypeReg**. Read-only. 80h=Type 0 multi-function device.
15:8	**LatencyTimer**. Read-only.
7:0	**CacheLineSize**. Read-only.

D0F2x2C Subsystem and Subvendor ID

Bits	Description
31:16	**SubsystemId**. Read-only. Value: 1577h.
15:0	**SubsystemVendorId**. Read-only. Value:1022h.

D0F2x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr**. Read-only. Reset: 40h.

D0F2x3C Interrupt Line

Bits	Description
31:16	Reserved.

15:8	**InterruptPin**. Read-only. Reset: 01h. This field indicates the INTx line used to generate legacy interrupts. Bits: Description 00h: Reserved. 01h: INTA. 02h: INTB. 03h: INTC. 04h: INTD. FFh-05h: Reserved.
7:0	**InterruptLine**. Read-write. Reset: 0. This field is read/write for software compatibility. It controls no hardware.

D0F2x40 IOMMU Capability

Bits	Description
31:29	Reserved.
28	**IommuCapExt**. Read-only. Reset: 1. Indicate IOMMU capability is extended to include D0F2x54 [IOMMU Miscellaneous Information Register High]. See D0F2x7C[CapExtW].
27	**IommuEfrSup**. Read-only. Reset: 1. 1=Indicates IOMMUx30 [Extended Feature Low] is supported. 0=IOMMUx30 is Reserved.
26	**IommuNpCache**. Read-only. Reset: 0. 1=Indicates that the IOMMU caches page table entries that are marked as not present. When this bit is set, software must issue an invalidate after any change to a PDE or PTE. 0=Indicates that the IOMMU caches only page table entries that are marked as present. When this bit is clear, software must issue an invalidate after any change to a PDE or PTE marked present before the change.
25	**IommuHtTunnelSup**. Read-only. Reset: 0.
24	**IommuIoTlbsup**. Read-only. Reset: 1. Indicates support for remote IOTLBs.
23:19	**IommuCapRev**. Read-only. Reset: 1. Specifies the IOMMU interface revision.
18:16	**IommuCapType**. Read-only. Reset: 3h. Specifies the layout of the Capability Block as an IOMMU capability block.
15:8	**IommuCapPtr**. Read-only. Reset: 64h. Indicates the location of the next capability block.
7:0	**IommuCapId**. Read-only. Reset: Fh. Indicates a Secure Device capability block.

D0F2x44 IOMMU Base Address Low

Bits	Description
31:14	**IommuBaseAddr[31:14]: IOMMU base address bits[31:14]**. IF (D0F2x44[IommuEnable]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. IommuBaseAddr[63:14] = {D0F2x48[IommuBaseAddr[63:32]], IommuBaseAddr[31:14]}. IommuBaseAddr[63:14] specifies the base address of the IOMMU memory mapped control registers. In order to use the IOMMU event counters, IommuBaseAddr[18:14] must be 0_0000b.

13:1	Reserved.
0	**IommuEnable**. Read; Write-1-only. Reset: 0. 1=IOMMU accepts memory accesses to the address specified in IommuBaseAddr[63:14]. When this bit is set, all IOMMU RW capability registers in PCI configuration space are locked.

D0F2x48 IOMMU Base Address High

Bits	Description
31:0	**IommuBaseAddr[63:32]: IOMMU base address bits[63:32]**. See: D0F2x44[IommuBase-Addr[31:14]].

D0F2x4C IOMMU Range

Bits	Description
31:24	**IommuLastDevice**. Read-only. Reset: 0. Indicates device and function number of the last integrated device associated with the IOMMU.
23:16	**IommuFirstDevice**. Read-only. Reset: 0. Indicates device and function number of the first integrated device associated with the IOMMU.
15:8	**IommuBusNumber**. Read-only. Reset: 0. Indicates the bus number that IommuLastDevice and Iom-muFirstDevice reside on.
7	**IommuRngValid**. Read-only. Reset: 0. 1=The IommuBusNumber, IommuFirstDevice, and Iommu-LastDevice fields are valid. Although the register contents are valid, software is encouraged to use IO topology information. 0=Software must use IO topology information.
6:5	Reserved.
4:0	**IommuUnitId**. Read-only. Reset: 0.

D0F2x50 IOMMU Miscellaneous Information Register

Bits	Description
31:27	**IommuMsiNumPpr**. Read-only. Reset: 0. This field must indicate which MSI vector is used for the interrupt message generated by the IOMMU for the peripheral page service request log when IOMMUx30[PprSup] == 1. This field must be 0 when IOMMUx30[PprSup] == 0. For MSI there can be only one IOMMU so this field must be 0. This interrupt is not remapped by the IOMMU.
26:23	Reserved.
22	**IommuHtAtsResv**. IF (D0F2x44[IommuEnable]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 1=The link Address Translation address range for ATS responses is reserved and cannot be translated by the IOMMU. 0=The Address Translation address range can be translated by the IOMMU.

21:15	**IommuVaSize**. Read-only. Reset: 40h. This field must indicate the size of the maximum virtual address processed by the IOMMU. The value is the (unsigned) binary log of the maximum address size. Bits — Description 19h-00h — Reserved. 20h — 32 bits. 27h-21h — Reserved. 28h — 40 bits. 2Fh-29h — Reserved. 30h — 48 bits. 3Fh-31h — Reserved. 40h — 64 bits. 7Fh-41h — Reserved.
14:8	**IommuPaSize**. Read-only. Reset: 30h. This field must indicate the size of the maximum physical address generated by the IOMMU. The value is the (unsigned) binary log of the maximum address size. Bits — Description 27h-00h — Reserved. 28h — 40 bits. 29h — Reserved. 30h — 48 bits. 7Fh-31h — Reserved.
7:5	**IommuGvaSize**. Read-only. Reset: 010b. Indicates the size of the maximum guest virtual address processed by the IOMMU. Bits — Description 001b-000b — Reserved. 010b — 48 bits. 111b-011b — Reserved.
4:0	**IommuMsiNum**. Read-only. Reset: 0. Indicates the MSI vector used for interrupt messages generated by the IOMMU.

D0F2x54 IOMMU Miscellaneous Information Register High

Bits	Description
31:5	Reserved.
4:0	**IommuMsiNumGa**. Read-only. Reset: 0. Indicates the MSI vector used for guest AVIC interrupts generated by the IOMMU. This field must be 0 when IOMMUx30[GaSup] == 0. This interrupt is not remapped by the IOMMU.

D0F2x64 IOMMU MSI Capability Register

Bits	Description
31:24	Reserved.
23	**Msi64En**. Read-only. Reset: 1. 1=64-bit MSI addressing is supported.

22:20	**MsiMultMessEn**. Read-write. Reset: 0. Specifies the number of MSI messages assigned to this function.
19:17	**MsiMultMessCap**. Read-only. Reset: 0. Specifies the number of MSI messages requested by this function.
16	**MsiEn**. Read-write. Reset: 0. 1=Enables MSI for this function and causes legacy interrupts to be disabled.
15:8	**MsiCapPtr**. Read-only. Reset: 74h. Pointer to the next capability register offset.
7:0	**MsiCapId**. Read-only. Reset: 5h. Indicates that this is the MSI capability.

D0F2x68 IOMMU MSI Address Low

Bits	Description
31:2	**MsiAddr[31:2]**. Read-write. Reset: 0. This register specifies the lower address bits used to issue MSI messages.
1:0	Reserved.

D0F2x6C IOMMU MSI Address High

Bits	Description
31:0	**MsiAddr[63:32]**. Read-write. Reset: 0. This register specifies the upper address bits used to issue MSI messages.

D0F2x70 IOMMU MSI Data

Bits	Description
31:16	Reserved.
15:0	**MsiData**. Read-write. Reset: 0. This register specifies the data issued with MSI messages.

D0F2x74 IOMMU MSI Mapping Capability

Bits	Description
31:27	**MsiMapCapType**. Read-only. Reset: 15h. Indicates the MSI Mapping Capability.
26:18	Reserved.
17	**MsiMapFixd**. Read-only. Reset: 1. 1=MSI interrupt mapping range is not programmable.
16	**MsiMapEn**. Read-only. Reset: 1. Always set to 1 to indicate that the MSI Mapping Capability is always enabled.
15:8	**MsiMapCapPtr**. Read-only. Reset: 0. Points to the next capability list item.
7:0	**MsiMapCapId**. Read-only. Reset: 8h. Indicates a link capability list item.

D0F2x7C IOMMU Control

Bits	Description
31:14	Reserved.
13	**CapExtW**. Read-write. Reset: 1. This field sets the value of D0F2x40[IommuCapExt].
12:10	**MsiMultMessCapW**. Read-write. Reset: 0h. BIOS: 2h.
9	**EfrSupW**. Read-write. Reset: 1. This field sets the value of D0F2x40[IommuEfrSup].
8	**IoTlbsupW**. Read-write. Reset: 1. This field sets the value of D0F2x40[IommuIoTlbsup].
7:4	**MinorRevIdW**. Read-write. Reset: 0. This field sets the value of D0F2x08[RevID[3:0]].
3	Reserved.
2:0	**InterruptPinW**. Read-write. Reset: 1. This field sets the value of D0F2x3C[InterruptPin].

D0F2x80 IOMMU MMIO Control Low

Bits	Description
31:30	**GLXSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx30[GlxSup].
29:28	**EventFW**. Read-write. Reset: 2. This field sets the value of IOMMUx30[EventF].
27:26	Reserved.
25:24	**PprFSupW**. Read-write. Reset: 2. This field sets the value of IOMMUx30[PprF].
23:21	**GamSupW**. Read-write. Reset:1. This field sets the value of IOMMUx30[GamSup].
20:13	Reserved.
12	**UsSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx34[UsSup]. BIOS: This field should not be re-programmed to 0.
11:10	**HatsW**. Read-write. Reset: 2h. This field sets the value of IOMMUx30[HATS].
9	**PcSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx30[PcSup].
8	Reserved.
7	**GaSupW**. Read-write. Reset:1. This field sets the value of IOMMUx30[GaSup].
6	**IaSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx30[IaSup].
5	Reserved.
4	**GtSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx30[GtSup].
3	**NxSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx30[NxSup]. BIOS: Program D0F2xFC_x07_L1i[3:0][ForceNoExePerm] = 1 when NxSupW == 0.
2	Reserved.
1	**PprSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx30[PprSup].
0	**PrefSupW**. Read-write. Reset: 0. BIOS: 0. This field sets the value of IOMMUx30[PrefSup].

D0F2x84 IOMMU MMIO Control High

Bits	Description
31:13	Reserved.
12:11	**MarcNumSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx34[MarcNum].
10	**BlockStopMarkSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx34[BlockStop-MarkSup].
9	**PprAutoRespSupW**. Read-write. Reset: 1. This field sets the value of IOMMUx34[PprAutoResp-Sup].
8	**PprOverflowEarlySupW**. Read-write. Reset: 1. This field sets the value of IOMMUx34[PprOver-flowEarlySup].
7:6	**DTESegW**. Read-write. Reset: 3. This field sets the value of IOMMUx34[DTESeg].
5:4	Reserved.
3:0	**PasMaxW**. Read-write. Reset: Fh. This field sets the value of IOMMUx34[PasMax].

D0F2x88 IOMMU Range Control

The fields in this register set the values of the corresponding fields in D0F2x4C.

Bits	Description
31:24	**LastDeviceW**. Read-write. Reset: 0.
23:16	**FirstDeviceW**. Read-write. Reset: 0.
15:8	**BusNumberW**. Read-write. Reset: 0.
7	**RngValidW**. Read-write. Reset: 0.
6:0	Reserved.

D0F2x8C IOMMU DSFX Control

The fields in this register set the values of the corresponding fields in IOMMUx138.

Bits	Description
31:24	**RevID**. Read-write. Reset: 0. This field sets the value in IOMMUx138[RevID], IOMMUx140[RevID], and IOMMUx148[RevID].
23:0	**DsfxSup**. Read-write. Reset: 0. This field sets the value in IOMMUx138[DsfxSup].

D0F2x90 IOMMU DSSX Control

Bits	Description
31:24	Reserved.
23:0	**DssxStatusSet**. Read-write. Reset: 0. This field sets the value of IOMMUx148[DssxStatus].

D0F2x94 IOMMU MARC Status

Bits	Description
31:13	Reserved.
12	**MarcGlobalEnStatus**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx1C[MarcEn] has been updated.
11	**MarcBase3Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx248 and/or IOMMUx24C has been udpated.
10	**MarcReloc3Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx250 and/or IOMMUx254 has been udpated.
9	**MarcLen3Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx258 and/or IOMMUx25C has been udpated.
8	**MarcBase2Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx230 and/or IOMMUx234 has been udpated.
7	**MarcReloc2Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx238 and/or IOMMUx23C has been udpated.
6	**MarcLen2Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx240 and/or IOMMUx244 has been udpated.
5	**MarcBase1Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx218 and/or IOMMUx21C has been udpated.
4	**MarcReloc1Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx220 and/or IOMMUx224 has been udpated.
3	**MarcLen1Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx228 and/or IOMMUx22C has been udpated.
2	**MarcBase0Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx200 and/or IOMMUx204 has been udpated.
1	**MarcReloc0Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx208 and/or IOMMUx20C has been udpated.
0	**MarcLen0Status**. Read; Write-1-to-clear. Reset: 0. Indicate IOMMUx210 and/or IOMMUx214 has been udpated.

D0F2xF0 IOMMU L2 Config Index

The index/data pair registers, D0F2xF0 and D0F2xF4 are used to access the registers at D0F2xF4_x[FF:00]. To access any of these registers, the address is first written into the index register, D0F2xF0, and then the data is read from or written to the data register, D0F2xF4. See 2.12.1 [IOMMU Configuration Space].

Bits	Description
31:9	Reserved.
8	**L2cfgWrEn**. Read-write. Reset: 0.
7:0	**L2cfgIndex**. Read-write. Reset: 0.

D0F2xF4 IOMMU L2 Config Data

IF (D0F2xF0[L2cfgWrEn]) THEN Read-write. ELSE Read-only. Reset: 0000_0000h. See D0F2xF0. Address: D0F2xF0[L2cfgIndex].

Bits	Description
31:0	**L2cfgData**.

D0F2xF4_x00 L2_PERF_CNTL_0

Bits	Description
31:24	**L2PerfCountUpper1**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 1.
23:16	**L2PerfCountUpper0**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 0.
15:8	**L2PerfEvent1**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 1.
7:0	**L2PerfEvent0**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 0.

D0F2xF4_x01 L2_PERF_COUNT_0

Bits	Description
31:0	**L2PerfCount0**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 0.

D0F2xF4_x02 L2_PERF_COUNT_1

Bits	Description
31:0	**L2PerfCount1**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 1.

D0F2xF4_x03 L2_PERF_CNTL_1

Bits	Description
31:24	**L2PerfCountUpper3**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 3.
23:16	**L2PerfCountUpper2**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 2.
15:8	**L2PerfEvent3**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 3.
7:0	**L2PerfEvent2**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 2.

D0F2xF4_x04 L2_PERF_COUNT_2

Bits	Description
31:0	**L2PerfCount2**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 2.

D0F2xF4_x05 L2_PERF_COUNT_3

Bits	Description
31:0	**L2PerfCount3**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 3.

D0F2xF4_x08 L2_STATUS_0

Bits	Description
31:0	**L2STATUS0**. Read-only. Reset: 0. Internal IOMMU L2A status.

D0F2xF4_x0C L2_CONTROL_0

Bits	Description
31:24	**IFifoClientPriority**. Read-write. Reset: 0. Each bit of this register controls whether the corresponding L1 client is arbitrated as high priority or not. Not all implementations use all of the priority bits due to a lower number of clients versus the register width.
23:20	**IFifoBurstLength**. Read-write. Reset: 1. Sets the burst length when arbitrating between clients coming into the L2.
19	Reserved.
18	**FLTCMBPriority**. Read-write. Reset: 0. 0=Round-robin arbitration between cache responses and table-walker responses at the fault combiner. 1=Table-walker responses always win arbitration at the fault combiner.
17:12	**IFifoCMBCredits**. Read-write. Reset: 4h. Controls the initial number of credits for the ififo to fault/CMB interface. Credits are loaded whenever the register value changes. This register may only be programmed when IOMMU is not enabled to preserve correct operation.
11	**SIDEPTEOnAddrTransExcl**. Read-write. Reset: 0. 0=Caches return DTE to L1 on an address translation exclusion range access. 1=Caches return PTE to L1 on an address translation exclusion range access.
10	**SIDEPTEOnUntransExcl**. Read-write. Reset: 0. 0=Caches return DTE to L1 on an untranslated exclusion range access. 1=Caches return PTE to L1 on an untranslated exclusion range access.
9:4	**IFifoTWCredits**. Read-write. Reset: 4h. Controls the initial number of credits for the ififo to TW interface. Credits are loaded whenever the register changes value. This register may only be programmed when IOMMU is not enabled to preserve correct operation.
3	**DTCHitVZeroOrIVZero**. Read-write. Reset: 0. 0=A DTE is refetched if a DTE with V == 0 for a memory request or IV == 1 for an interrupt request is hit in the DTC. 1=A DTE is not refetched if a DTE with V == 0 for a memory request or IV == 1 for an interrupt request is hit in the DTC. This DTE is used.
2	**AllowL1CacheATSRsp**. Read-write. Reset: 0. 0=L2 does not allow L1 to cache responses to ATS address translation requests. 1=L2 allows L1 to cache responses to ATS address translation requests.
1	**AllowL1CacheVZero**. Read-write. Reset: 0. 0=L2 does not allow L1 to cache DTEs where V == 0. 1=L2 allows L1 to cache DTEs where V == 1. L1 stores IR and IW as if they are both set to 1.
0	**PTCAddrTransReqCheck**. Read-write. Reset: 0. 0=Address translation requests do not check the PTC. 1=Address translation requests check the PTC.

D0F2xF4_x0D L2_CONTROL_1

Bits	Description
31:24	**PerfThreshold**. Read-write. Reset: 0. Fifo threshold level used to calculate certain performance counter values.
23:17	Reserved.
16	**SeqInvBurstLimitEn**. Read-write. Reset: 1. Enable stalling L2 requests to allow invalidation cycles to make forward progress based upon SeqInvBurstLimitInv and SeqInvBurstLimitL2Req.
15:8	**SeqInvBurstLimitL2Req**. Read-write. Reset: 8h. Sets the number of consecutive IOMMU L2 requests to perform when doing sequential invalidation. Regular L2 and invalidation requests alternates access to the main L2 caches based upon SeqInvBurstLimitInv and SeqInvBurstLimitL2Req.
7:0	**SeqInvBurstLimitInv**. Read-write. Reset: 8h. Sets the number of consecutive invalidation requests to perform when doing sequential invalidation. Regular L2 and invalidation requests alternates access to the main L2 caches based upon SeqInvBurstLimitInv and SeqInvBurstLimitL2Req.

D0F2xF4_x10 L2_DTC_CONTROL

Bits	Description
31:28	**DTCEntries**. Read-only. Reset: 6h. The number of entries in the DTC is indicated as 2^DTCEntries.
27:24	Reserved.
23:16	**DTCWays**. Read-only. Reset: 2h. Indicates the number of ways in the DTC.
15	**DTCParitySupport**. Read-only. Reset: 1. 0=The DTC does not support parity protection. 1=The DTC supports parity protection.
14	Reserved.
13	**DTCBypass**. Read-write. Reset: 0. When set, all requests bypass the DTC.
12:11	Reserved.
10	**DTCSoftInvalidate**. Read-write. Reset: 0. Software may write this register to 1 to invalidate all entries in the DTC.
9:8	**DTCInvalidationSel**. Read-write. Reset: 0. BIOS: 10b. Selects the DTC invalidation algorithm. Bits — Description 00b — Invalidate the entire DTC. 01b — Fast imprecise invalidation. 10b — Sequential precise invalidation. 11b — Partial sequential precise invalidation.
7:5	Reserved.
4	**DTCParityEn**. Read-write. Reset: 0. Enable parity protection of the DTC.
3	**DTCLRUUpdatePri**. Read-write. Reset: 0. 0=Reads update replacement state bits when there is a simultaneous read and write to the same DTC index. 1=Writes update replacement state bits when there is a simultaneous read and write to the same DTC index.
2	Reserved.
1:0	**DTCReplacementSel**. Read-write. Reset: 1. Selects the DTC replacement algorithm. Implementation may not support all replacement algorithms.

D0F2xF4_x11 L2_DTC_HASH_CONTROL

Bits	Description
31:16	**DtcAddressMask**. Read-write. Reset: 0. This field is a bit-wise AND mask that selects which bits from the untranslated interrupt {MT[2:0],Vector} are used to index into the DTC.
15:11	Reserved.
10	**DtcAltHashEn**. Read-write. Reset: 0. Enable alternative algorithm for generating hash index into the DTC.
9	Reserved.
8:5	**DTCBusBits**. Read-write. Reset: 3h. Set the number of bus bits to use when using ReqID to form the DTC address. The following equation must be satisified. Func_bits + Dev_Bits + Bus_Bits <= log2(DTC entries / DTC associativity).
4:2	**DTCDevBits**. Read-write. Reset: 0. Set the number of device bits to use when using ReqID to form the DTC address.
1:0	**DTCFuncBits**. Read-write. Reset: 2h. Set the number of function bits to use when using ReqID to form the DTC address.

D0F2xF4_x12 L2_DTC_WAY_CONTROL

Bits	Description
31:16	**DTCWayAccessDisable**. Read-write. Reset: 0.
15:0	**DTCWayDisable**. Read-write. Reset: 0. Each bit in this register disables a way in the DTC when set to 1. An implementation may have less than 32 ways. The entire cache may be disabled by setting the DTCWays lower bits of this register.

D0F2xF4_x14 L2_ITC_CONTROL

Bits	Description
31:28	**ITCEntries**. Read-only. Reset: 6h. The number of entries in the ITC is indicated as 2^ITCEntries.
27:24	Reserved.
23:16	**ITCWays**. Read-only. Reset: 2. Indicates the number of ways in the ITC.
15	**ITCParitySupport**. Read-only. Reset: 1. 0=The ITC does not support parity protection. 1=The ITC supports parity protection.
14	Reserved.
13	**ITCBypass**. Read-write. Reset: 0. When set, all requests bypass the ITC.
12:11	Reserved.
10	**ITCSoftInvalidate**. Read-write. Reset: 0. Software may write this register to 1 to invalidate all entries in the ITC.

9:8	**ITCInvalidationSel**. Read-write. Reset: 0. BIOS: See 2.12.2. Selects the ITC invalidation algorithm. Bits: Description 00b: Invalidate the entire ITC. 01b: Fast imprecise invalidation. 10b: Sequential precise invalidation. 11b: Partial sequential precise invalidation.
7:5	Reserved.
4	**ITCParityEn**. Read-write. Reset: 0. Enable parity protection of the ITC.
3	**ITCLRUUpdatePri**. Read-write. Reset: 0. 0=Reads update replacement state bits when there is a simultaneous read and write to the same ITC index. 1=Writes update replacement state bits when there is a simultaneous read and write to the same ITC index.
2	Reserved.
1:0	**ITCReplacementSel**. Read-write. Reset: 1. Selects the ITC replacement algorithm. Implementation may not support all replacement algorithms.

D0F2xF4_x15 L2_ITC_HASH_CONTROL

Bits	Description
31:16	**ITCAddressMask**. Read-write. Reset: 0h. This register is a bit-wise AND mask that selects which bits from the untranslated interrupt {MT[2:0],Vector} are used to index into the ITC.
15:11	Reserved.
10	**ItcAltHashEn**. Read-write. Reset: 0. Enable alternative algorithm for generating hash index into the ITC.
9	Reserved.
8:5	**ITCBusBits**. Read-write. Reset: 3h. Set the number of bus bits to use when using ReqID to form the ITC address. The following equation must be satisified. Func_bits + Dev_Bits + Bus_Bits <= log2(ITC entries / ITC associativity).
4:2	**ITCDevBits**. Read-write. Reset: 0. Set the number of device bits to use when using ReqID to form the ITC address.
1:0	**ITCFuncBits**. Read-write. Reset: 2h. Set the number of function bits to use when using ReqID to form the ITC address.

D0F2xF4_x16 L2_ITC_WAY_CONTROL

Bits	Description
31:16	**ITCWayAccessDisable**. Read-write. Reset: 0.
15:0	**ITCWayDisable**. Read-write. Reset: 0. Each bit in this register disables a way in the ITC when set to 1. An implementation may have less than 32 ways. The entire cache may be disabled by setting the ITCWays lower bits of this register.

D0F2xF4_x18 L2_PTC_A_CONTROL

Bits	Description
31:28	**PTCAEntries**. Read-only. Reset: 8h. The number of entries in the PTC A sub-cache is indicated as 2^PTCAEntries.
27:24	Reserved.
23:16	**PTCAWays**. Read-only. Reset: 4h. Indicates the number of ways in the PTC A sub-cache.
15	**PTCAParitySupport**. Read-only. Reset: 1. 0=The PTC A sub-cache does not support parity protection. 1=The PTC A sub-cache supports parity protection.
14	Reserved.
13	**PTCABypass**. Read-write. Reset: 0. When set, all requests bypass the PTC A sub-cache.
12	Reserved.
11	**PTCA2MMode**. Read-write. Reset: 0. When set, the PTC A sub-cache stores 2M pages instead of 4K pages.
10	**PTCASoftInvalidate**. Read-write. Reset: 0. Software may write this register to 1 to invalidate all entries in the PTC A sub-cache.
9:8	**PTCAInvalidationSel**. Read-write. Reset: 0. BIOS: See 2.12.2. Selects the PTC A sub-cache invalidation algorithm. Bits — Description 00b — Invalidate the entire PTC A sub-cache. 01b — Fast imprecise invalidation. 10b — Sequential precise invalidation. 11b — Partial sequential precise invalidation.
7:5	Reserved.
4	**PTCAParityEn**. Read-write. Reset: 0. Enable parity protection of the PTC A sub-cache.
3	**PTCALRUUpdatePri**. Read-write. Reset: 0. 0=Reads update replacement state bits when there is a simultaneous read and write to the same PTCA index. 1=Writes update replacement state bits when there is a simultaneous read and write to the same PTCA index.
2	Reserved.
1:0	**PTCAReplacementSel**. Read-write. Reset: 1. Selects the PTC A sub-cache replacement algorithm. Implementation may not support all replacement algorithms.

D0F2xF4_x19 L2_PTC_A_HASH_CONTROL

Bits	Description
31:16	**PTCAAddressMask**. Read-write. Reset: 0. This register is a bit-wise AND mask that selects which virtual address bits are used to index into the PTC A sub-cache.
15:11	Reserved.
10	**PtcAltHashEn**. Read-write. Reset: 0. Enable alternative algorithm for generating hash index into the PTC.
9	Reserved.

8:5	**PTCABusBits**. Read-write. Reset: 3h. Set the number of bus bits to use when using ReqID to form the PTC A sub-cache address. The following equation must be satisified: FuncBits + DevBits + BusBits <= log2(PTC A sub-cache entries / PTC A sub-cache associativity).
4:2	**PTCADevBits**. Read-write. Reset: 0. Set the number of device bits to use when using ReqID to form the PTC A sub-cache address.
1:0	**PTCAFuncBits**. Read-write. Reset: 2h. Set the number of function bits to use when using ReqID to form the PTC A sub-cache address.

D0F2xF4_x1A L2_PTC_A_WAY_CONTROL

Bits	Description
31:16	**PTCAWayAccessDisable**. Read-write. Reset: 0.
15:0	**PTCAWayDisable**. Read-write. Reset: 0. Each bit in this register disables a way in the PTC A sub-cache when set to 1. An implementation may have less than 32 ways. The entire cache may be disabled by setting the PTCAWays lower bits of this register.

D0F2xF4_x1C L2_PTC_B_CONTROL

Bits	Description
31:28	**PTCBEntries**. Read-only. Reset: 8h. The number of entries in the PTC B sub-cache is indicated as 2^PTCBEntries.
27:24	Reserved.
23:16	**PTCBWays**. Read-only. Reset: 4h. Indicates the number of ways in the PTC B sub-cache.
15	**PTCBParitySupport**. Read-only. Reset: 1. 0=The PTC B sub-cache does not support parity protection. 1=The PTC B sub-cache supports parity protection.
14	Reserved.
13	**PTCBBypass**. Read-write. Reset: 0. 1=All requests bypass the PTC B sub-cache.
12	Reserved.
11	**PTCB2MMode**. Read-write. Reset: 0. 1=The PTC B sub-cache stores 2M pages instead of 4K pages.
10	**PTCBSoftInvalidate**. Read-write. Reset: 0. 1=Invalidate all entries in the PTC B sub-cache.
9:8	**PTCBInvalidationSel**. Read-write. Reset: 0. Selects the PTC B sub-cache invalidation algorithm. Bits — Description 00b — Invalidate the entire PTC B sub-cache. 01b — Fast imprecise invalidation. 10b — Sequential precise invalidation. 11b — Partial sequential precise invalidation.
7:5	Reserved.
4	**PTCBParityEn**. Read-write. Reset: 0. Enable parity protection of the PTC B sub-cache.
3	**PTCBLRUUpdatePri**. Read-write. Reset: 0. 0=Reads update replacement state bits when there is a simultaneous read and write to the same PTCB index. 1=Writes update replacement state bits when there is a simultaneous read and write to the same PTCB index.

2	Reserved.
1:0	**PTCBReplacementSel**. Read-write. Reset: 1. Selects the PTC B sub-cache replacement algorithm. Implementation may not support all replacement algorithms.

D0F2xF4_x1D L2_PTC_B_HASH_CONTROL

Bits	Description
31:16	**PTCBAddressMask**. Read-write. Reset: 0. This register is a bit-wise AND mask that selects which virtual address bits are used to index into the PTC B sub-cache.
15:9	Reserved.
8:5	**PTCBBusBits**. Read-write. Reset: 3h. Set the number of bus bits to use when using ReqID to form the PTC B sub-cache address. The following equation must be satisified. FuncBits + DevBits + BusBits <= log2(PTC B sub-cache entries / PTC B sub-cache associativity).
4:2	**PTCBDevBits**. Read-write. Reset: 0. Set the number of device bits to use when using ReqID to form the PTC B sub-cache address.
1:0	**PTCBFuncBits**. Read-write. Reset: 2h. Set the number of function bits to use when using ReqID to form the PTC B sub-cache address.

D0F2xF4_x1E L2_PTC_B_WAY_CONTROL

Bits	Description
31:16	**PTCBWayAccessDisable**. Read-write. Reset: 0.
15:0	**PTCBWayDisable**. Read-write. Reset: 0. Each bit in this register disables a way in the PTC A sub-cache when set to 1. An implementation may have less than 32 ways. The entire cache may be disabled by setting the PTCBWays lower bits of this register.

D0F2xF4_x20 L2_CREDIT_CONTROL_2

Bits	Description
31:28	Reserved.
27:24	**PprLoggerCredits**. Read-write. Reset: 4h. PPR log buffer credit override value.
23	**FCELOverride**. Read-write. Reset: 0. Changing this register from 0 to 1 overrides the FCEL credit counter with FCELCredits. This should only be performed when the IOMMU is idle.
22	Reserved.
21:16	**FCELCredits**. Read-write. Reset: 0. FCEL credit override value.
15	**FLTCMBOverride**. Read-write. Reset: 0. Changing this register from 0 to 1 overrides the FLTCMB credit counter with FLTCMBCredits. This should only be performed when the IOMMU is idle.
14	Reserved.
13:8	**FLTCMBCredits**. Read-write. Reset: 0. FLTCMB credit override value.
7	**QUEUEOverride**. Read-write. Reset: 0. Changing this register from 0 to 1 overrides the QUEUE credit counter with QUEUECredits. This should only be performed when the IOMMU is idle.

6	Reserved.
5:0	**QUEUECredits**. Read-write. Reset: 0. QUEUE credit override value.

D0F2xF4_x22 L2A_UPDATE_FILTER_CNTL

Bits	Description
31:5	Reserved.
4:1	**L2aUpdateFilterRdlatency**. Read-write. Reset: 3h. When L2aUpdateFilterBypass is 0, assume the invalidation read has completed in the number of clock cycles specified by this field.
0	**L2aUpdateFilterBypass**. Read-write. Reset: 1. 1=Disable duplicate update filtering. 0=Enable the dropping of updates that are already in the L2aUpdateFilter or in the destination L2A cache.

D0F2xF4_x30 L2_ERR_RULE_CONTROL_3

Bits	Description
31:4	**ERRRuleDisable3**. Read-write. Reset: 0. Each bit in this register disables an error detection rule in the IOMMU.
3:1	Reserved.
0	**ERRRuleLock1**. Read; write-once. Reset: 0. BIOS: See 2.12.2. Setting this register bit locks the error detection rule set in ERRRuleDisable3, D0F2xF4_x31[ERRRuleDisable4] and D0F2xF4_x32[ERRRuleDisable5].

D0F2xF4_x31 L2_ERR_RULE_CONTROL_4

Bits	Description
31:0	**ERRRuleDisable4**. Read-write. Reset: 0. Each bit in this register disables an error detection rule in the IOMMU.

D0F2xF4_x32 L2_ERR_RULE_CONTROL_5

Bits	Description
31:0	**ERRRuleDisable5**. Read-write. Reset: 0. Each bit in this register disables an error detection rule in the IOMMU.

D0F2xF4_x33 L2_L2A_CK_GATE_CONTROL

Bits	Description
31:8	Reserved.

7:6	**CKGateL2AStop**. Read-write. Reset: 01b. Bits Description 00b Allow 2 clock cycles delay before stopping the clocks when clkready deasserts. 01b Allow 4 clock cycles delay before stopping the clocks when clkready deasserts. 10b Allow 8 clock cycles delay before stopping the clocks when clkready deasserts. 11b Allow 16 clock cycles delay before stopping the clocks when clkready deasserts.
5:4	**CKGateL2ALength**. Read-write. Reset: 01b. Bits Description 00b Allow 128 clock cycles delay before stopping the clocks when idle asserts. 01b Allow 256 clock cycles delay before stopping the clocks when idle asserts. 10b Allow 512 clock cycles delay before stopping the clocks when idle asserts. 11b Allow 1024 clock cycles delay before stopping the clocks when idle asserts.
3	**CKGateL2ASpare**. Read-write. Reset: 0. Spare bit.
2	**CKGateL2ACacheDisable**. Read-write. Reset: 1. BIOS: 0. 1=Disable the gating of the L2A upper cache ways.
1	**CKGateL2ADynamicDisable**. Read-write. Reset: 1. BIOS: 0. 1=Disable the gating of the L2A dynamic clock branch.
0	**CKGateL2ARegsDisable**. Read-write. Reset: 1. BIOS: 0. 1=Disable the gating of the L2A register clock branch.

D0F2xF4_x34 L2_L2A_PGSIZE_CONTROL

Bits	Description
31:4	Reserved.
3:2	**L2aregHostPgsize**. Read-write. Reset: 0. BIOS: See 2.12.2.
1:0	**L2aregGstPgsize**. Read-write. Reset: 0. BIOS: See 2.12.2.

D0F2xF4_x35 L2_L2A_PGMEM_CONTROL_1

Bits	Description
31:3	Reserved.
2	**L2aregSDEn**. Read-write. Reset: 0. Enable shut down power gating for L2A memories. This field must be programmed to 0.
1	**L2aregDSEn**. Read-write. Reset: 0. Enable deep sleep power gating for L2A memories. This field must be programmed to 0.
0	**L2aregLSEn**. Read-write. Reset: 0. Enable light sleep power gating for L2A memories. This field must be programmed to 0.

D0F2xF4_x36 L2_L2A_PGMEM_CONTROL_2

Bits	Description
31:0	**L2aregLSThres**. Read-write. Reset: 64h. Threshold value for L2A memories to enter Light Sleep.

D0F2xF4_x37 L2_L2A_PGMEM_CONTROL_3

Bits	Description
31:0	**L2aregDSThres**. Read-write. Reset: 64h. Threshold value for L2A memories to enter Deep Sleep.

D0F2xF4_x38 L2_L2A_PGMEM_CONTROL_4

Bits	Description
31:0	**L2aregSDThres**. Read-write. Reset: 44Ch. Threshold value for L2A memories to enter shut down.

D0F2xF4_x3B IOMMU_PGFSM_CONFIG

Bits	Description
31:28	**RegAddr**. Read-write. Reset: 0. Indicate the register address for write or read operation.
27	**SrbmOverride**. Read-write. Reset: 0.
26:14	Reserved.
13	**PGRead**. Read-write. Reset: 0. Indicate a read operation is performed.
12	**PGWrite**. Read-write. Reset: 0. Indicate a write operation is performed.
11	**P2Select**. Read-write. Reset: 0. Indicate the power up or down is for P2 domain.
10	**P1Select**. Read-write. Reset: 0. Indicate the power up or down is for P1 domain.
9	**PowerUp**. Read-write. Reset: 0. Request power up.
8	**PowerDown**. Read-write. Reset: 0. Request power down.
7:0	**FsmAddr**. Read-write. Reset: 0. Indicate the address of the PGFSM. FFh=broadcast to all PGFSM.

D0F2xF4_x3C IOMMU_PGFSM_WRITE

Bits	Description
31:0	**WriteValue**. Read-write. Reset: 0.

D0F2xF4_x3D IOMMU_PGFSM_READ

Bits	Description
31:24	Reserved.
23:0	**ReadValue**. Read-only. Reset: 0.

D0F2xF4_x3E L2_PG_CNTL_0

Bits	Description
31:0	**IpPgThres**. Read-write. Reset: 3E8h.

D0F2xF4_x3F L2_PG_CNTL_1

Bits	Description
31:0	**IpPgPwrDnDelayCnt**. Read-only. Reset: 64h. Indicate the power down delay.

D0F2xF4_x40 L2_PG_CNTL_2

Bits	Description
31:0	**IpPgPwrUpDelayCnt**. Read-only. Reset: 64h. Indicate the power up delay.

D0F2xF4_x41 L2_PG_CNTL_3

Bits	Description
31:2	Reserved.
1	**IpPgBusy**. Read-only. Reset: 0.
0	**IpPgEn**. Read-write. Reset: 0. BIOS:0.

D0F2xF4_x48 L2_STATUS_1

Bits	Description
31:0	**L2STATUS1**. Read-only. Reset: 0. Internal IOMMU L2B status.

D0F2xF4_x49 L2_SB_LOCATION

Bits	Description
31:16	**SbLocatedCore**. Read-write. Reset: 0. Specifies the core location of the FCH.
15:0	**SbLocatedPort**. Read-write. Reset: 0. BIOS: See 2.12.2 [IOMMU Initialization]. Specifies the port location of the FCH. Bit PortLocation [0] FCH is located on port A of the corresponding PCIe® core or internal FCH. [1] FCH is located on port B of the corresponding PCIe core. [2] FCH is located on port C of the corresponding PCIe core. [3] FCH is located on port D of the corresponding PCIe core. [15:4] Reserved.

D0F2xF4_x4C L2_CONTROL_5

Bits	Description
31:12	Reserved.
11:10	Reserved.

9:8	**GstPartialPtcCntrl**. Read-write. Reset: 0. BIOS: 11b.
7	**FC2AltMode**. Read-write. Reset: 0. 0=FC2 primary flow-control mode. 1=FC2 alternate flow-control mode.
6	**FC3Dis**. Read-write. Reset: 0. 0=FC3 flow-control loop is enabled. 1=FC3 flow-control look is disabled.
5	**FC2Dis**. Read-write. Reset: 0. 0=FC2 flow-control loop is enabled. 1=FC2 flow-control look is disabled.
4	**DTCUpdateVZeroIVOne**. Read-write. Reset: 0. 0=DTEs with V == 0 and IV == 1 are not cached in the DTC. 1=DTEs with V == 0 and IV == 1 are cached in the DTC.
3	**DTCUpdateVOneIVZero**. Read-write. Reset: 0. 0=DTEs with V == 1 and IV == 0 are not cached in the DTC. 1=DTEs with V == 1 and IV == 0 are cached in the DTC.
2	**FC1Dis**. Read-write. Reset: 0. 0=FC1 flow control loop enabled. 1=FC1 flow control loop disabled.
1	**PTCAddrTransReqUpdate**. Read-write. Reset: 1. 0=PTEs from address translation requests are not cached. 1=PTEs from address translation requests are cached in the L2 according to the Cache bit in the DTE.
0	**QueueArbFBPri**. Read-write. Reset: 1. 0=Requests in the miss queue and the feedback queue are arbitrated in a round-robin manner. 1=Requests in the feedback queue are given priority over requests in the miss queue.

D0F2xF4_x4D L2_CONTROL_6

Bits	Description
31:24	**Perf2Threshold**. Read-write. Reset: 0. Fifo threshold level used to calculate certain performance counter values.
23:17	Reserved.
16	**SeqInvBurstLimitEn**. Read-write. Reset: 1. Enable stalling PDC requests to allow invalidation cycles to make forward progress based upon SeqInvBurstLimitInv and SeqInvBurstLimitPDCReq.
15:8	**SeqInvBurstLimitPDCReq**. Read-write. Reset: 8h. Sets the number of consecutive IOMMU PDC requests to perform when doing sequential invalidation. PDC and invalidation requests alternates access to the PDC based upon SeqInvBurstLimitInv and SeqInvBurstLimitPDCReq.
7:0	**SeqInvBurstLimitInv**. Read-write. Reset: 8h. Sets the number of consecutive invalidation requests to perform when doing sequential invalidation. PDC and invalidation requests alte rnates access to the PDC based upon SeqInvBurstLimitInv and SeqInvBurstLimitPDCReq.

D0F2xF4_x50 L2_PDC_CONTROL

Bits	Description
31:28	**PDCEntries**. Read-only. Reset: 8h. Indicates the number of entries in the PDC is indicated as 2^PDCEntries.
27:24	Reserved.
23:16	**PDCWays**. Read-only. Reset: 4h. Indicates the number of ways in the PDC.
15	**PDCParitySupport**. Read-only. Reset: 1. 0=The PDC does not support parity protection. 1=The PDC supports parity protection.

14	Reserved.
13	**PDCBypass**. Read-write. Reset: 0. When set, all requests bypass the PDC. This prevents the multiple issue of requests and increases maximum rate of requests to the table-walker.
12	**PDCSearchDirection**. Read-write. Reset: 0. 0=Search PDC from higher levels down. 1=Search PDC from lower levels up.
11	Reserved.
10	**PDCSoftInvalidate**. Read-write. Reset: 0. Software may write this register to 1 to invalidate all entries in the PDC.
9:8	**PDCInvalidationSel**. Read-write. Reset: 0. BIOS: See 2.12.2. Selects the PDC invalidation algorithm. Bits — Description 00b — Invalidate the entire PDC. 01b — Fast imprecise invalidation. 10b — Sequential precise invalidation. 11b — Partial sequential precise invalidation.
7:5	Reserved.
4	**PDCParityEn**. Read-write. Reset: 0. Enable parity protection of the PDC if the device supports parity.
3	**PDCLRUUpdatePri**. Read-write. Reset: 0. 0=Reads update replacement state bits when there is a simultaneous read and write to the same PDC index. 1=Writes update replacement state bits when there is a simultaneous read and write to the same PDC index.
2	Reserved.
1:0	**PDCReplacementSel**. Read-write. Reset: 1. Selects the PDC replacement algorithm. Implementation may not support all replacement algorithms.

D0F2xF4_x51 L2_PDC_HASH_CONTROL

Bits	Description
31:16	**PDCAddressMask**. Read-write. Reset: 0. This register is a bit-wise AND mask that selects which virtual address bits are used to index into the PDC.
15:11	Reserved.
10	**PdcAltHashEn**. Read-write. Reset: 0. 1=Enable alternative algorithm for generating hash index into the PDC.
9	**PDCUpperLvlAddrHash**. Read-write. Reset: 1. When set to 1, the PDC cache index is partially formed using the XOR of the LSBs of virtual address bits for all levels greater than or equal to the stored/searched level.
8	**PDCLvlHash**. Read-write. Reset: 1. When set to 1, the PDE level is used as part of the hash for the cache index.
7:6	Reserved.
5:0	**PDCDomainBits**. Read-write. Reset: 7h. Selects the number of domain bits to use as part of the index into the PDC.

D0F2xF4_x52 L2_PDC_WAY_CONTROL

Bits	Description
31:16	**PDCWayAccessDisable**. Read-write. Reset: 0.
15:0	**PDCWayDisable**. Read-write. Reset: 0. Each bit in this register disables a way in the PDC when set to 1. An implementation may have less than 32 ways. The entire cache may be disabled by setting the PDCWays lower bits of this register.

D0F2xF4_x53 L2B_UPDATE_FILTER_CNTL

Bits	Description
31:5	Reserved.
4:1	**L2bUpdateFilterRdlatency**. Read-write. Reset: 3h. When L2bUpdateFilterBypass is 0, assume the invalidation read has completed in the number of clock cycles specified by this field.
0	**L2bUpdateFilterBypass**. Read-write. Reset: 1. 1=Disable duplicate update filtering. 0=Enable the dropping of updates that are already in the l2b_update_filter or in the PDC.

D0F2xF4_x54 L2_TW_CONTROL

Bits	Description
31:19	Reserved.
18	**TwContWalkPErrDis**. Read-write. Reset: 0. 0=Continue walking tables on a write permission error. 1=Stop walking tables on a write permission error.
17	**Twfilter64bDis**. Read-write. Reset: 0.
16	**TwfilterDis**. Read-write. Reset: 0.
15	Reserved.
14:12	**TWPrefetchRange**. Read-write. Reset: 1. BIOS: 7h. Selects the number of pages to prefetch.
11	**TWPTEOnAddrTransExcl**. Read-write. Reset: 0. 0=Table walker returns DTE to L1 on an address translation exclusion range access. 1=Table walker returns PTE to L1 on an address translation exclusion range access.
10	**TWPTEOnUntransExcl**. Read-write. Reset: 0. 0=Table walker returns DTE to L1 on an untranslated exclusion range access. 1=Table walker returns PTE to L1 on an untranslated exclusion range access.
9	**TWPrefetchOnly4KDis**. Read-write. Reset: 0. 1=Allow non-4K pages to be prefetched. 0=Only 4K pages are prefetched.
8	**TWPrefetchEn**. Read-write. Reset: 0. BIOS: 1. Enable prefetching in the table-walker.
7	Reserved.
6	**TWForceCoherent**. Read-write. Reset: 0. 1=Table-walker always genereates coherent requests. The DTE SD bit is ignored when this bit is set to 1.
5:0	**TWQueueLimit**. Read-write. Reset: 10h. Limit the number of outstanding table-walker requests.

D0F2xF4_x56 L2_CP_CONTROL

Bits	Description
31:16	**CPRdDelay**. Read-write. Reset: 0. Command processor read delay.
15:3	Reserved.
2	**CPFlushOnInv**. Read-write. Reset: 1. BIOS: 0. 1=Command processor flushes out old requests on every invalidation command. 0=No flush is performed during invalidations.
1	**CPFlushOnWait**. Read-write. Reset: 0. BIOS: 1. 1=Command processor flushes out old requests on completion wait. 0=No flush is performed on completion wait.
0	**CPPrefetchDis**. Read-write. Reset: 0. 1=Command processor fetches and executes only one command at a time. 0=Command processor prefetches available commands into its internal storage.

D0F2xF4_x57 L2_CP_CONTROL_1

Bits	Description
31:16	Reserved.
15:0	**CPL1Off[15:0]**. Read-write. Reset: 0. Each bit in this register indicates to the IOMMU command processor that a corresponding L1 TLB is inaccessible due to static clock or power gating. System software is responsible for programming this register. Bit Definition Bit Definition [0] PPx40 [3] IOAGR [1] Reserved [15-4] Reserved [2] BIF

D0F2xF4_x58 IOMMU_L2_GUEST_ADDR_CNTRL

Bits	Description
31:24	Reserved.
23:0	**IommuL2GuestAddrMask**. Read-write. Reset: 0.

D0F2xF4_x6A L2_INT_CONTROL

Bits	Description
31:4	Reserved.
3	**IntGaOrderEn**. Read-write. Reset: 1. 1=Enable ordering between interrupts and guest virtual APIC log writes.
2	**IntPPROrderEn**. Read-write. Reset: 1. 1=Enable ordering between interrupts and PPR log writes.
1	**IntCPOrderEn**. Read-write. Reset: 1. 1=Enable ordering between interrupts and command processor writes.
0	**IntEventOrderEn**. Read-write. Reset: 1. 1=Enable ordering between interrupts and event log writes.

D0F2xF4_x70 L2_CREDIT_CONTROL_0

Bits	Description
31	**DTEOverride**. Read-write. Reset: 0. 1=Override the DTE credit counter with DTECredits. Setting should only be performed when the IOMMU is idle.
30	Reserved.
29:24	**DTECredits**. Read-write. Reset: 2h. DTE credit override value.
23	**FC3Override**. Read-write. Reset: 0. 1=Override the FC3 credit counter with FC3Credits. Setting should only be performed when the IOMMU is idle.
22	Reserved.
21:16	**FC3Credits**. Read-write. Reset: 0. FC3 credit override value.
15	**FC2Override**. Read-write. Reset: 0. 1=Override the FC2 credit counter with FC2Credits. This should only be performed when the IOMMU is idle.
14	Reserved.
13:8	**FC2Credits**. Read-write. Reset: 0. FC2 credit override value.
7	**FC1Override**. Read-write. Reset: 0. 1=Override the FC1 credit counter with FC1Credits. Setting should only be performed when the IOMMU is idle.
6	Reserved.
5:0	**FC1Credits**. Read-write. Reset: 0. FC1 credit override value.

D0F2xF4_x71 L2_CREDIT_CONTROL_1

Bits	Description
31:24	Reserved.
23:20	**PprMcifCredits**. Read-write. Reset: 4h. PPR logger credit override value.
19:16	**CpPrefetchCredits**. Read-write. Reset: 4h. Command processor prefetch credit override value.
15	**TWELOverride**. Read-write. Reset: 0. 1=Override the TWEL credit counter with TWELCredits. Setting should only be performed when the IOMMU is idle.
14	Reserved.
13:8	**TWELCredits**. Read-write. Reset: 4h. TWEL credit override value.
7	**PDTIEOverride**. Read-write. Reset: 0. 1=Override the PDTIE credit counter with PDTIECredits. Setting should only be performed when the IOMMU is idle.
6	Reserved.
5:0	**PDTIECredits**. Read-write. Reset: 4h. PDTIE credit override value.

D0F2xF4_x78 L2_MCIF_CONTROL

Bits	Description
31:29	Reserved.

28:24	**MCIFBaseWriteDataCredits**. Read-write. Reset: 8h. Sets the number of base-channel write data credits between the IOMMU L2 and the HTIU/ORB. Software must ensure no traffic is on this data path while programming this register.
23:21	Reserved.
20:16	**MCIFBaseWriteHdrCredits**. Read-write. Reset: 8h. Sets the number of base-channel write header credits between the IOMMU L2 and the HTIU/ORB. Software must ensure no traffic is on this data path while programming this register.
15:13	Reserved.
12:8	**MCIFIsocReadCredits**. Read-write. Reset: 8h. Sets the number of isoc-channel read credits between the IOMMU L2 and the HTIU/ORB. Software must ensure no traffic is on this data path while programming this register.
7:5	Reserved.
4:0	**MCIFBaseReadCredits**. Read-write. Reset: 8h. Sets the number of base-channel read credits between the IOMMU L2 and the HTIU/ORB. Software must ensure no traffic is on this data path while programming this register.

D0F2xF4_x80 L2_ERR_RULE_CONTROL_0

Bits	Description
31:4	**ERRRuleDisable0**. Read-write. Reset: 0. Each bit in this register disables an error detection rule in the IOMMU.
3:1	Reserved.
0	**ERRRuleLock0**. Read-write. Reset: 0. BIOS: See 2.12.2. This register is write-once. Setting this register bit locks the error detection rule set in ERRRuleDisable0/1/2.

D0F2xF4_x81 L2_ERR_RULE_CONTROL_1

Bits	Description
31:0	**ERRRuleDisable1**. Read-write. Reset: 0. Each bit in this register disables an error detection rule in the IOMMU.

D0F2xF4_x82 L2_ERR_RULE_CONTROL_2

Bits	Description
31:0	**ERRRuleDisable2**. Read-write. Reset: 0. Each bit in this register disables an error detection rule in the IOMMU.

D0F2xF4_x90 L2_L2B_CK_GATE_CONTROL

Bits	Description
31:8	Reserved.

7:6	**CKGateL2BStop**. Read-write. Reset: 01b. Bits Description 00b Allow 2 clock cycles delay before stopping the clocks when clkready deasserts. 01b Allow 4 clock cycles delay before stopping the clocks when clkready deasserts. 10b Allow 8 clock cycles delay before stopping the clocks when clkready deasserts. 11b Allow 16 clock cycles delay before stopping the clocks when clkready deasserts.
5:4	**CKGateL2BLength**. Read-write. Reset: 01b. Bits Description 00b Allow 128 clock cycles delay before stopping the clocks when idle asserts. 01b Allow 256 clock cycles delay before stopping the clocks when idle asserts. 10b Allow 512 clock cycles delay before stopping the clocks when idle asserts. 11b Allow 1024 clock cycles delay before stopping the clocks when idle asserts.
3	**CKGateL2BCacheDisable**. Read-write. Reset: 0. 1=Disable the gating of the L2B upper cache ways.
2	**CKGateL2BMiscDisable**. Read-write. Reset: 1. 1=Disable the gating of the L2B miscellaneous clock branch.
1	**CKGateL2BDynamicDisable**. Read-write. Reset: 1. BIOS: 0. 1=Disable the gating of the L2B dynamic clock branch.
0	**CKGateL2BRegsDisable**. Read-write. Reset: 1. BIOS: 0. 1=Disable the gating of the L2B TLB register clock branch.

D0F2xF4_x92 PPR_CONTROL

Bits	Description
31:17	Reserved.
16	**PprIntcoallesceEn**. Read-write. Reset: 0. BIOS: See 2.12.2.
15:8	**PprIntreqdelay**. Read-write. Reset: 0. BIOS: See 2.12.2.
7:0	**PprInttimedelay**. Read-write. Reset: 0. BIOS: See 2.12.2.

D0F2xF4_x94 L2_L2B_PGSIZE_CONTROL

Bits	Description
31:4	Reserved.
3:2	**L2bregHostPgsize**. Read-write. Reset: 0. BIOS: See 2.12.2.
1:0	**L2bregGstPgsize**. Read-write. Reset: 0. BIOS: See 2.12.2.

D0F2xF4_x95 L2_L2B_PGMEM_CONTROL_1

Bits	Description
31:3	Reserved.
2	**L2bregSDEn**. Read-write. Reset: 0. BIOS: 0. Enable shut down power gating for L2B memories.

1	**L2bregDSEn**. Read-write. Reset: 0. Enable deep sleep power gating for L2B memories. This field must be programmed to 0.
0	**L2bregLSEn**. Read-write. Reset: 0. Enable light sleep power gating for L2B memories. This field must be programmed to 0.

D0F2xF4_x96 L2_L2B_PGMEM_CONTROL_2

Bits	Description
31:0	**L2bregLSThres**. Read-write. Reset: 64h. Threshold value for L2B memories to enter Light Sleep.

D0F2xF4_x97 L2_L2B_PGMEM_CONTROL_3

Bits	Description
31:0	**L2bregDSThres**. Read-write. Reset: 64h. Threshold value for L2B memories to enter Deep Sleep.

D0F2xF4_x98 L2_L2B_PGMEM_CONTROL_4

Bits	Description
31:0	**L2bregSDThres**. Read-write. Reset: 44Ch. Threshold value for L2B memories to enter shut down.

D0F2xF4_x99 L2_PERF_CNTL_2

Bits	Description
31:24	**L2PerfCountUpper5**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 5.
23:16	**L2PerfCountUpper4**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 4.
15:8	**L2PerfEvent5**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 5.
7:0	**L2PerfEvent4**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 4.

D0F2xF4_x9A L2_PERF_COUNT_4

Bits	Description
31:0	**L2PerfCount4**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 4.

D0F2xF4_x9B L2_PERF_COUNT_5

Bits	Description
31:0	**L2PerfCount5**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 5.

D0F2xF4_x9C L2_PERF_CNTL_3

Bits	Description
31:24	**L2PerfCountUpper7**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 7.
23:16	**L2PerfCountUpper6**. Read-only. Reset: 0. Upper 8 bits of IOMMU L2 performance counter 6.
15:8	**L2PerfEvent7**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 7.
7:0	**L2PerfEvent6**. Read-write. Reset: 0. Selects the IOMMU L2 performance counter event for counter 6.

D0F2xF4_x9D L2_PERF_COUNT_6

Bits	Description
31:0	**L2PerfCount6**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 6.

D0F2xF4_x9E L2_PERF_COUNT_7

Bits	Description
31:0	**L2PerfCount7**. Read-only. Reset: 0. Lower 32 bits of IOMMU L2 performance counter 7.

D0F2xF8 IOMMU L1 Config Index

The index/data pair registers, D0F2xF8 and D0F2xFC are used to access the registers at D0F2xFC_x[FFFF:0000]_L1[3:0]. To access any of these registers, the address is first written into the index register, D0F2xF8, and then the data is read from or written to the data register, D0F2xFC.

See 2.12.1 [IOMMU Configuration Space]. There are various L1s in the IOMMU. Registers in the L1 indexed space have one instance per L1 denoted by _L1i[x] where x=D0F2xF8[L1cfgSel]. The syntax for this register type is described by the following example:

- D0F2xFC_x00 refers to all instances of the D0F2xFC_x00 registers.
- D0F2xFC_x00_L1i[0] refers to the D0F2xFC_x00 register instance for the PPx40 L1.

Bits	Description
31	**L1cfgEn**. Read-write. Reset: 0. 1=Enable writes to D0F2xFC.
30:20	Reserved.
19:16	**L1cfgSel**. Read-write. Reset: 0. This field selects one of the following L1s to access. Bits / Definition: 0h PPx40; 1h Reserved; 2h BIF; 3h IOAGR; Fh-4h Reserved
15:0	**L1cfgIndex**. Read-write. Reset: 0.

D0F2xFC IOMMU L1 Config Data

IF (D0F2xF8[L1cfgEn]) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0000_0000h. See D0F2xF8. Address: D0F2xF8[L1cfgIndex].

Bits	Description
31:0	**L1cfgData**.

D0F2xFC_x00_L1i[3:0] L1_PERF_CNTL

Bits	Description
31:24	**L1PerfCountHi1**. Read-only. Reset: 0. Read back of perf counter 1 bits[39:32].
23:16	**L1PerfCountHi0**. Read-only. Reset: 0. Read back of perf counter 0 bits[39:32].
15:8	**L1PerfEvent1**. Read-write. Reset: 0. Perf counter event 1.
7:0	**L1PerfEvent0**. Read-write. Reset: 0. Perf counter event 0.

D0F2xFC_x01_L1i[3:0] L1_PERF_COUNT_0

Bits	Description
31:0	**L1PerfCount0**. Read-only. Reset: 0. Read back of perf counter 0 bits[31:0].

D0F2xFC_x02_L1i[3:0] L1_PERF_COUNT_1

Bits	Description
31:0	**L1PerfCount1**. Read-only. Reset: 0. Read back of perf counter 1 bits[31:0].

D0F2xFC_x07_L1i[3:0] L1_DEBUG_1

Bits	Description
31:18	Reserved.
17	**L1NwEn**. Read-write. Reset: 0. BIOS: 1. 1=Enable NW bit on ATS requests.
16:15	Reserved.
14	**AtsPhysPageOverlapDis**. Read-write. Reset: 0. BIOS: 1. 1=Prevent physical page overlap for ATS responses.
13	Reserved.
12	**AtsSeqNumEn**. Read-write. Reset: 0. BIOS: 1. 1=Enable logging of ATS sequence number.
11	**SpecReqFilterEn**. Read-write. Reset: 0. BIOS: 1. 1=Filter special requests in L1 work queue.
10:1	Reserved.
0	**PhantomFuncDis**. Read-write. Reset: 0. BIOS: See 2.12.2. 1=Disable phantom function support.

D0F2xFC_x09_L1i[3:0] L1_SB_LOCATION

Bits	Description
31:16	**SbLocatedCore**. Read-write. Reset: 0. BIOS: See 2.12.2 [IOMMU Initialization]. Specifies the core location of the FCH.
15:0	**SbLocatedPort**. Read-write. Reset: 0. BIOS: See 2.12.2 [IOMMU Initialization]. Specifies the port location of the FCH. Bit — PortLocation [0] FCH is located on port A of the corresponding PCIe® core or internal FCH. [1] FCH is located on port B of the corresponding PCIe core. [2] FCH is located on port C of the corresponding PCIe core. [3] FCH is located on port D of the corresponding PCIe core. [15:4] Reserved.

D0F2xFC_x0C_L1i[3:0] L1_CNTRL_0

Bits	Description
31	Reserved.
30:28	**L1VirtOrderQueues**. Read-write. Reset: 0. BIOS: See 2.12.2. This field controls number of virtual queues in the L1 work queue. Bits — Description 0h — 1 1h — 2 2h — 4 3h — 8 4h — 16 7h-5h — Reserved
27:24	**L1Entries**. Read-only; Updated-by-hardware. Reset: 0. This field specifies the number of entries in each L1 cache as 2^L1 entries.
23:22	Reserved.
21:20	**L1Banks**. Read-only; Updated-by-hardware. Reset: 1. This field specifies number of caches in L1.
19:14	Reserved.
13:8	**L2Credits**. Read-write. Reset: 4h. This field controls credits for L1 to L2 interface.
7:6	Reserved.
5	**ReplacementSel**. Read-write. Reset: 0.
4	Reserved.
3	**CacheiwOnly**. Read-write. Reset: 1. 1=Cache write only pages in L1.
2	**CacheirOnly**. Read-write. Reset: 1. 1=Cache read only pages in L1.
1	**FragmentDis**. Read-write. Reset: 0. 1=Disable variable page size support in L1 cache - only 4K pages.
0	**UnfilterDis**. Read-write. Reset: 0. 1=Disable unfiltering in L1 write queue of aborted L2 requests.

D0F2xFC_x0D_L1i[3:0] L1_CNTRL_1

Bits	Description
31:30	Reserved.
29	**Untrans2mFilteren**. Read-write. Reset: 0. Enable filtering of requests on a 2M boundry instead of 4K.
28	**PretransNovaFilteren**. Read-write. Reset: 0. When set, VA is not used for filtering pretrans requests.
27	**L1CacheSelInterleave**. Read-write. Reset: 0. When set causes cache updates to toggle between multiple caches.
26	**L1CacheSelReqid**. Read-write. Reset: 0. When set, allows the reqid to be used in hashing between multiple L1 caches.
25:23	**SelectTimeoutPulse**. Read-write. Reset: 0.
22	**L1CacheInvAllEn**. Read-write. Reset: 0. Enables invalidation of entire cache when invalidation command is sent.
21	**L1orderEn**. Read-write. Reset: 0. Enables strict ordering of all requests through L1.
20	**SndFilterDis**. Read-write. Reset: 0. Disables filtering of requests to L2.
19	**AtsNobufferInsert**. Read-write. Reset: 0. Disables buffering of read completion data when inserting ats responses.
18:14	**WqEntrydis**. Read-write. Reset: 0. Value indicates how many cache entries in L1 to disable.
13	**BlockL1Dis**. Read-only. Reset: 0.
12	**L1DTEDis**. Read-write. Reset: 0. Disables L1 caching of DTE.
11	**L1ParityEn**. Read-write. Reset: 0.
10	**L1CacheParityEn**. Read-write. Reset: 0. Enables forced miss of L1 cache due to failed parity check.
9	**CacheByPass**. Read-write. Reset: 0. Enables L1 cache bypass.
8	**VOQXorMode**. Read-write. Reset: 0.
7	Reserved.
6:4	**VOQFuncBits**. Read-write. Reset: 0.
3	Reserved.
2:0	**VOQPortBits**. Read-write. Reset: 0. !000b=Enable virtual queue hashing using port id, controls number of bits to use from port id for hashing.

D0F2xFC_x0E_L1i[3:0] L1_CNTRL_2

Bits	Description
31:28	Reserved.
27:20	**MsiHtRsvIntVector**. Read-write. Reset: 00h. This field defines the interrupt vector used when an MSI interrupt is received that has a reserved DM field.
19:12	**MsiHtRsvIntDestination**. Read-write. Reset: FFh. This field defines the interrupt destination used when an MSI interrupt is received that has a reserved DM field.
11	Reserved.

10	**MsiHtRsvIntDM**. Read-write. Reset: 0. Defines the interrupt destination mode when an MSI interrupt is received that has a reserved DM field.
9	**MsiHtRsvIntRqEio**. Read-write. Reset: 0. Specifies the RQEOI state when an MSI interrupt is received that has a reserved DM field.
8:6	**MsiHtRsvIntMt**. Read-write. Reset: 011b. Specifies the message type used when an MSI interrupt is received that has a reserved DM field.
5:3	Reserved.
2	**L1AbrtAtsDis**. Read-write. Reset: 0. 1=Disable abort of ats requests when IOMMU is disabled.
1	**MsiToHtRemapDis**. Read-write. Reset: 0. 1=Disable mapping of MSI to link interrupts.
0	Reserved.

D0F2xFC_x0F_L1i[3:0] L1_CNTRL_3

Bits	Description
31:0	**AtsTlbinvPulseWidth**. Read-write. Reset: C350h. Specifies the pulse width of the ats invalidation counters.

D0F2xFC_x10_L1i[3:0] L1_BANK_SEL_0

Bits	Description
31:16	Reserved.
15:0	**L1cachebanksel0**. Read-write. Reset: 1. Specifies value is used to determine the virtual address bit that selects between the 2 banks of the L1 cache (if present). The bank is selected by bitwise ANDing this register against virtual address bits[19:12] and XORing the result.

D0F2xFC_x11_L1i[3:0] L1_BANK_DISABLE_0

Bits	Description
31:14	Reserved.
13:8	**L1cachelinedis1**. Read-write. Reset: 0. Sets the number of cache entries to disable in cache 1.
7:6	Reserved.
5:0	**L1cachelinedis0**. Read-write. Reset: 0. Sets the number of cache entries to disable in cache 0.

D0F2xFC_x20_L1i[3:0] L1_WQ_STATUS_0

Table 79: Valid Values for D0F2xFC_x20_L1i[3:0]

Bits	Description
0h	Idle.
1h	Wait_L1.
2h	Wait_L2.

Table 79: Valid Values for D0F2xFC_x20_L1i[3:0] (Continued)

3h	Sending special request to L2.
4h	Waiting for completion of special request.
5h	Done.

Bits	Description
31:30	Reserved.
29:27	**EntryStatus9**. See: EntryStatus0.
26:24	**EntryStatus8**. See: EntryStatus0.
23:21	**EntryStatus7**. See: EntryStatus0.
20:18	**EntryStatus6**. See: EntryStatus0.
17:15	**EntryStatus5**. See: EntryStatus0.
14:12	**EntryStatus4**. See: EntryStatus0.
11:9	**EntryStatus3**. See: EntryStatus0.
8:6	**EntryStatus2**. See: EntryStatus0.
5:3	**EntryStatus1**. See: EntryStatus0.
2:0	**EntryStatus0**. Read-only. Reset: 0. See: Table 79 [Valid Values for D0F2xFC_x20_L1i[3:0]].

D0F2xFC_x21_L1i[3:0] L1_WQ_STATUS_1

Bits	Description
31:30	Reserved.
29:27	**EntryStatus19**. See: EntryStatus10.
26:24	**EntryStatus18**. See: EntryStatus10.
23:21	**EntryStatus17**. See: EntryStatus10.
20:18	**EntryStatus16**. See: EntryStatus10.
17:15	**EntryStatus15**. See: EntryStatus10.
14:12	**EntryStatus14**. See: EntryStatus10.
11:9	**EntryStatus13**. See: EntryStatus10.
8:6	**EntryStatus12**. See: EntryStatus10.
5:3	**EntryStatus11**. See: EntryStatus10.
2:0	**EntryStatus10**. Read-only. Reset: 0. See: Table 79 [Valid Values for D0F2xFC_x20_L1i[3:0]].

D0F2xFC_x22_L1i[3:0] L1_WQ_STATUS_2

Bits	Description
31:30	Reserved.
29:27	**EntryStatus29**. See: EntryStatus20.
26:24	**EntryStatus28**. See: EntryStatus20.

23:21	**EntryStatus27**. See: EntryStatus20.
20:18	**EntryStatus26**. See: EntryStatus20.
17:15	**EntryStatus25**. See: EntryStatus20.
14:12	**EntryStatus24**. See: EntryStatus20.
11:9	**EntryStatus23**. See: EntryStatus20.
8:6	**EntryStatus22**. See: EntryStatus20.
5:3	**EntryStatus21**. See: EntryStatus20.
2:0	**EntryStatus20**. Read-only. Reset: 0. See: Table 79 [Valid Values for D0F2xFC_x20_L1i[3:0]].

D0F2xFC_x23_L1i[3:0] L1_WQ_STATUS_3

Bits	Description
31:16	Reserved.
15:8	**InvalidationStatus**. Read-only. Reset: 0. Status of invalidation state machine.
7:6	Reserved.
5:3	**EntryStatus31**. See: EntryStatus30.
2:0	**EntryStatus30**. Read-only. Reset: 0. See: Table 79 [Valid Values for D0F2xFC_x20_L1i[3:0]].

D0F2xFC_x32_L1i[3:0] L1_CNTRL_4

Bits	Description
31:23	Reserved.
22	**Atomics64BOrderEn**. Read-write. Reset: 0. Enable port based cacheline ordering between DMA non-posted atomic requests.
21	**Posted64BOrderEn**. Read-write. Reset: 0. Enable port based cacheline ordering between DMA posted reqeusts.
20	**TaggedFenceEn**. Read-write. Reset: 0. 0=Tagged fence disable, L1 uses DMA target done interface to complete flushes. 1=L1 uses tagged fence to complete flushes.
19:17	**ForceDmaAttrLow**. Read-write. Reset: 0.
16	**DmaNpHaltDis**. Read-write. Reset: 0.
15:10	**DmaBufMaxNpCred**. Read-write. Reset: Fh.
9:4	**DmaBufCredits**. Read-write. Reset: 10h.
3	**TlpprefixerrEn**. Read-write. Reset: 0.
2	**TimeoutPulseExtEn**. Read-write. Reset: 0.
1	**AtsMultipleL1toL2En**. Read-write. Reset: 0. BIOS: See 2.12.2.
0	**AtsMultipleRespEn**. Read-write. Reset: 0. BIOS: See 2.12.2.

D0F2xFC_x33_L1i[3:0] L1_CLKCNTRL_0

Bits	Description
31	**L1L2ClkgateEn**. Read-write. Reset: 0. BIOS: 1.
30:12	Reserved.
11	**L1HostreqClkgateEn**. Read-write. Reset: 0. BIOS: 1.
10	**L1RegClkgateEn**. Read-write. Reset: 0. BIOS: 1.
9	**L1MemoryClkgateEn**. Read-write. Reset: 0. BIOS: 1.
8	**L1PerfClkgateEn**. Read-write. Reset: 0. BIOS: 1.
7	**L1DmaInputClkgateEn**. Read-write. Reset: 0. BIOS: 1.
6	**L1CpslvClkgateEn**. Read-write. Reset: 0. BIOS: 1.
5	**L1CacheClkgateEn**. Read-write. Reset: 0. BIOS: 1.
4	**L1DmaClkgateEn**. Read-write. Reset: 0. BIOS: 1.
3:2	Reserved.
1:0	**L1ClkgateLen**. Read-write. Reset: 0.

D0F2xFC_x34_L1i[3:0] L1_MEMPWRCNTRL_0

Bits	Description
31:24	**L1MempwrTimer2**. Read-write. Reset: Fh.
23:16	**L1MempwrTimer1**. Read-write. Reset: Fh.
15:8	**L1MempwrTimer0**. Read-write. Reset: Fh.
7:1	Reserved.
0	**L1MempwrEn**. Read-write. Reset: 0.

D0F2xFC_x35_L1i[3:0] L1_MEMPWRCNTRL_1

Bits	Description
31:8	Reserved.
7:0	**L1MempwrTimer3**. Read-write. Reset: Fh.

D0F2xFC_x36_L1i[3:0] L1_GUEST_ADDR_CNTRL

Bits	Description
31:8	**L1GuestAddrMsk**. Read-write. Reset: 0.
7:1	Reserved.
0	**L1CanonicalErrEn**. Read-write. Reset: 0.

D0F2xFC_x37_L1i[3:0] L1_FEATURE_SUP_CNTRL

Bits	Description
31:4	Reserved.
3:2	**L1DteSegW**. Read-write. Reset: 3.
1	**L1PprSup**. Read-write. Reset: 1.
0	**L1EfrSup**. Read-write. Reset: 1.

D0F2xFC_x38_L1i[3:0] L1_CNTRL_5

Bits	Description
31:25	Reserved.
24	**IocNpHaltEn**. Read-write. Reset: 1. 0=Disable non-posted request stalling. 1=Enable non-posted request stalling when IOC asserts np_halt to L1.
23:20	Reserved.
19:16	**HostReqCredits**. Read-write. Reset: 8h. Number of credits available at initial state for the host request interface from L1 to downstream client.
15	Reserved.
14	**VdmArbSizeEn**. Read-write. Reset: 0. Support for arbitrary size vendor defined messages.
13:8	**ClkOffWaitTime**. Read-write. Reset: 00_0011b. Programmable delay between clkready de-assert and lclk_idle assert (ie. turning off clk).
7:6	Reserved.
5:4	**HstCredits**. Read-write. Reset: 10b. Number of credits available for host response credit/debit interface.
3:0	**DmaCredits**. Read-write. Reset: 8h. Number of credits available for dma request credit/debit interface. D0F0x98_x02[OrbTxPgmemEn] must be cleared (pgmem disable) before programming the DMA credits.

D0F2xFC_x39_L1i[3:0] L1_PGMEM_CNTRL1

Bits	Description
31:3	Reserved.
2	**SDEn: Shut Down Enable**. Read-write. Reset: 0. 0=Disable power gated memories in L1 as it enters shut down state. 1=Enable power gated memories in L1 enters shut down state.
1	**DSEn: Deep Sleep Enable**. Read-write. Reset: 0. 0=Disable power gated memories in L1 as it enters Deep Sleep state. 1=Enable power gated memories in L1 enters Deep Sleep state.
0	**LSEn: Light Sleep Enable**. Read-write. Reset: 0. 0=Disable power gated memories in L1 as it enters Light Sleep state. 1=Enable power gated memories in L1 enters Light Sleep state.

D0F2xFC_x3A_L1i[3:0] L1_PGMEM_CNTRL2

Bits	Description
31:0	**LSThres: Light Sleep Threshold**. Read-write. Reset: 0. The number of LCLK cycles L1 stays in idle state before entering Light Sleep when D0F2xFC_x39_L1i[3:0][LSEn] == 1.

D0F2xFC_x3B_L1i[3:0] L1_PGMEM_CNTRL3

Bits	Description
31:0	**DSThres: Deep Sleep Threshold**. Read-write. Reset: 0. The number of LCLK cycles L1 stays in idle state before switching from previous power state to Deep Sleep when D0F2xFC_x39_L1i[3:0][DSEn] == 1.

D0F2xFC_x3C_L1i[3:0] L1_PGMEM_CNTRL4

Bits	Description
31:0	**SDThres: Shut DownThreshold**. Read-write. Reset: 0. The number of LCLK cycles L1 stays in idle state before switching from previous power state to shut down when D0F2xFC_x39_L1i[3:0][SDEn] == 1.

D0F2xFC_x3D_L1i[3:0] L1_SST_CNTRL0

Bits	Description
31:15	Reserved.
14	**HostReqCGEn**. Read-write. Reset: 0. 1=Clock gating enable on the host request branch of the sstunl_gnside.
13:0	Reserved.

D0F2xFC_x3E_L1i[3:0] L1_ATS_RESP_CNTRL0

Bits	Description
31	**L1AtsRespDlyEn**. Read-write. Reset: 0. Delay returning of unsuccessful ATS response when a PPR auto response returns prior to the ATS.
30:16	Reserved.
15:8	**L1AtsRespDlyTimer**. Read-write. Reset: 0. Values in 50 ns intervals.
7:0	**L1AtsRespAllowTimer**. Read-write. Reset: 0. Values in 50 ns intervals.

3.5 Device 1 Function 0 (Internal Graphics) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D1F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: Product-specific.
15:0	**VendorID: vendor ID**. Read-only. Value: 1002h.

D1F0x04 Status/Command Register

Reset: 0010_0000h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; Write-1-to-clear. 1=Poisoned PCIe® Transaction Layer Packet (TLP) received.
30	**SignaledSystemError: signaled system error**. Read; Write-1-to-clear. 1=A non-fatal or fatal error message was sent and SerrEn = 1.
29	**ReceivedMasterAbort: received master abort**. Read; Write-1-to-clear. 1=A completion with an unsupported request completion status was received.
28	**ReceivedTargetAbort: received target abort**. Read; Write-1-to-clear. 1=A completion with completer abort completion status was received.
27	**SignalTargetAbort: Signaled target abort**. Read-only.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**MasterDataPerr: master data parity error**. Read; Write-1-to-clear. 1=ParityErrorEn == 1 and either a poisoned completion was received or the device poisoned a write request.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	**UDFEn: UDF enable**. Read-only.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1=Capability list supported.
19	**IntStatus: interrupt status**. Read-only. 1=INTx interrupt message pending.
18:11	Reserved.
10	**IntDis: interrupt disable**. Read-write. 1=INTx interrupt messages generation disabled.
9	**FastB2BEn: fast back-to-back enable**. Read-only.
8	**SerrEn: System error enable**. Read-write. 1=Enables reporting of non-fatal and fatal errors detected.
7	**Stepping: Stepping control**. Read-only.
6	**ParityErrorEn: parity error response enable**. Read-write.
5	**PalSnoopEn: VGA palette snoop enable**. Read-only.

4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.
2	**BusMasterEn: bus master enable**. Read-write. 1=Memory and IO read and write request generation enabled.
1	**MemAccessEn: IO access enable**. Read-write. This bit controls if memory accesses targeting this device are accepted. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-write. This bit controls if IO accesses targeting this device are accepted. 1=Enabled. 0=Disabled.

D1F0x08 Class Code/Revision ID Register

Bits	Description
31:8	**ClassCode**. Value: 03_0000h.
7:0	**RevID: revision ID**. Value: Product-specific.

D1F0x0C Header Type Register

Reset: 0080_0000h.

Bits	Description
31:24	**BIST**. Read-only.
23:16	**HeaderTypeReg**. Read-only. The header type field indicates a header type 0 and that this is a multi-function device.
15:8	**LatencyTimer**. Read-only. These bits are fixed at their default value.
7:0	**CacheLineSize**. Read-write. This field specifies the system cache line size in units of double words.

D1F0x10 Graphic Memory Base Address

IF (D0F0xD4_x0109_14E2[StrapBifF064BarDisA]== 1) THEN Reset: 0000_0008h. ELSE Reset: 0000_000Ch. ENDIF.

Bits	Description
31:26	**BaseAddr[31:26]: base address**. Read-write. The amount of memory requested by the graphics memory BAR is controlled by D0F0xD4_x0109_1507[StrapBifMemApSizePin] and D0F0xD4_x0109_14E1[StrapBifMemApSize].
25:4	**BaseAddr[25:4]: base address**. Read-only.
3	**Pref: prefetchable**. Read-only. 1=Prefetchable memory region.
2:1	**Type: base address register type**. Read-only. Bits — Description 00b — 32-bit BAR 01b — Reserved 10b — 64-bit BAR 11b — Reserved
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

D1F0x14 Graphics Memory Base Address 64

Reset: 0000_0000h.

Bits	Description
31:0	**BaseAddr[63:32]: base address**. Read-write. This field is reserved if (D0F0xD4_x0109_14E2[StrapBifF064BarDisA] == 1).

D1F0x18 Graphics Doorbell Base Address

IF (D0F0xD4_x0109_14E2[StrapBifF064BarDisA] == 1) THEN Reset: 0000_0008h. ELSE Reset: 0000_000Ch. ENDIF. This register is reserved and reset is 0000_0000h if (D0F0xD4_x0109_14E1[StrapBifDoorbellBarDis] == 1).

Bits	Description
31:23	**BaseAddr[31:23]: base address**. Read-write.
22:4	**BaseAddr[22:4]: base address**. Read-only.
3	**Pref: prefetchable**. Read-only. 1=Prefetchable memory region.
2:1	**Type: base address register type**. Read-only. Bits Description 00b 32-bit BAR 01b Reserved 10b 64-bit BAR 11b Reserved
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

D1F0x1C Graphics Doorbell Base Address 64

Reset: 0000_0000h.

Bits	Description
31:0	**BaseAddr[63:32]: base address**. Read-write. This field is reserved if (D0F0xD4_x0109_14E1[StrapBifDoorbellBarDis] == 1 \|\| D0F0xD4_x0109_14E2[StrapBifF064BarDisA] == 1).

D1F0x20 Graphics IO Base Address

Reset: 0000_0000h. This register is called Base Address 4 if (D0F0xD4_x0109_14E2[StrapBifF064BarDisA] == 1).

Bits	Description
31:0	Reserved.

D1F0x24 Graphics Memory Mapped Registers Base Address

Reset: 0000_0000h.

Bits	Description
31:16	**BaseAddr[31:16]: base address**. Read-write. The amount of memory requested by the graphics memory mapped registers BAR is controlled by D0F0xD4_x0109_14E1[StrapBifRegApSize].
15:4	**BaseAddr[15:4]: base address**. Read-only.
3	**Pref: prefetchable**. Read-only. 0=Non-prefetchable memory region.
2:1	**Type: base address register type**. Read-only. 00b=32-bit BAR.
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

D1F0x2C Subsystem and Subvendor ID Register

Reset: 0000_0000h. This register can be modified through D1F0x4C

Bits	Description
31:16	**SubsystemID**. Read-only.
15:0	**SubsystemVendorID**. Read-only.

D1F0x30 Expansion ROM Base Address

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F0x34 Capabilities Pointer

Reset: 0000_0050h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Pointer to PM capability.

D1F0x3C Interrupt Line

Reset: 0000_01FFh.

Bits	Description
31:16	Reserved.
15:8	**InterruptPin: interrupt pin**. Read-only. This field identifies the legacy interrupt message the function uses.
7:0	**InterruptLine: interrupt line**. Read-write. This field contains the interrupt line routing information.

D1F0x4C Subsystem and Subvendor ID Mirror

Reset: 0000_0000h.

Bits	Description
31:16	**SubsystemID**. Read-write. This field sets the value in the corresponding field in D1F0x2C.
15:0	**SubsystemVendorID**. Read-write. This field sets the value in the corresponding field in D1F0x2C.

D1F0x50 Power Management Capability

Bits	Description
31:27	**PmeSupport**. Value: 0_0000b. Indicates that there is no PME support.
26	**D2Support: D2 support**. Value: 1. D2 is supported.
25	**D1Support: D1 support**. Value: 1. D1 is supported.
24:22	**AuxCurrent: auxiliary current**. Value: 0.
21	**DevSpecificInit: device specific initialization**. Value: 0. Indicates that there is no device specific initialization necessary.
20	Reserved.
19	**PmeClock**. Value: 0.
18:16	**Version: version**. Value: 011b.
15:8	**NextPtr: next pointer**. Value: 58h.
7:0	**CapID: capability ID**. Value: 01h. Indicates that the capability structure is a PCI power management data structure.

D1F0x54 Power Management Control and Status

Reset: 0000_0000h.

Bits	Description
31:24	**PmeData**. Read-only.
23	**BusPwrEn**. Read-only.
22	**B2B3Support**. Read-only. B states are not supported.
21:16	Reserved.
15	**PmeStatus: PME status**. Read-only.
14:13	**DataScale: data scale**. Read-only.
12:9	**DataSelect: data select**. Read-only.
8	**PmeEn: PME# enable**. Read-only.
7:4	Reserved.
3	**NoSoftReset: no soft reset**. Read-only. Software is required to re-initialize the function when returning from $D3_{hot}$.

2	Reserved.
1:0	**PowerState: power state**. Read-write. This 2-bit field is used both to determine the current power state of the root port and to set the root port into a new power state. Bits Definition 00b D0 10b-01b Reserved 11b $D3_{hot}$

D1F0x58 PCI Express® Capability

Bits	Description
31:30	Reserved.
29:25	**IntMessageNum: interrupt message number**. Value: 0. This field indicates which MSI vector is used for the interrupt message.
24	**SlotImplemented: Slot implemented**. Value: 0.
23:20	**DeviceType: device type**. Value: 9h.
19:16	**Version**. Value: 2h.
15:8	**NextPtr: next pointer**. Value: Product-specific.
7:0	**CapID: capability ID**. Value: 10h.

D1F0x5C Device Capability

Bits	Description
31:29	Reserved.
28	**FlrCapable: function level reset capability**. Value: 0.
27:26	**CapturedSlotPowerScale: captured slot power limit scale**. Value: 0.
25:18	**CapturedSlotPowerLimit: captured slot power limit value**. Value: 0.
17:16	Reserved.
15	**RoleBasedErrReporting: role-based error reporting**. Value: 1.
14:12	Reserved.
11:9	**L1AcceptableLatency: endpoint L1 Acceptable Latency**. Value: 111b.
8:6	**L0SAcceptableLatency: endpoint L0s Acceptable Latency**. Value: 110b.
5	**ExtendedTag: extended tag support**. Value: 1. 8-bit tag support.
4:3	**PhantomFunc: phantom function support**. Value: 0. No phantom functions supported.
2:0	**MaxPayloadSupport: maximum supported payload size**. Value: 000b. 128-byte max payload size.

D1F0x60 Device Control and Status

Reset: 0000_0810h.

Bits	Description
31:22	Reserved.

21	**TransactionsPending: transactions pending**. Read-only.
20	**AuxPwr: auxiliary power**. Read-only.
19	**UsrDetected: unsupported request detected**. Read; Write-1-to-clear. 1=Unsupported request received.
18	**FatalErr: fatal error detected**. Read; Write-1-to-clear. 1=Fatal error detected.
17	**NonFatalErr: non-fatal error detected**. Read; Write-1-to-clear. 1=Non-fatal error detected.
16	**CorrErr: correctable error detected**. Read; Write-1-to-clear. 1=Correctable error detected.
15	**BridgeCfgRetryEn: bridge configuration retry enable**. Read-only.
14:12	**MaxRequestSize: maximum request size**. Read-only.
11	**NoSnoopEnable: enable no snoop**. Read-write. 1=The device is permitted to set the No Snoop bit in requests.
10	**AuxPowerPmEn: auxiliary power PM enable**. Read-only. This capability is not implemented.
9	**PhantomFuncEn: phantom functions enable**. Read-only. Phantom functions are not supported.
8	**ExtendedTagEn: extended tag enable**. Read-write. 1=8-bit tag request tags. 0=5-bit request tag.
7:5	**MaxPayloadSize: maximum supported payload size**. Read-only. 000b=Indicates a 128-byte maximum payload size.
4	**RelaxedOrdEn: relaxed ordering enable**. Read-write. 1=The device is permitted to set the Relaxed Ordering bit.
3	**UsrReportEn: unsupported request reporting enable**. Read-write. 1=Enables signaling unsupported requests by sending error messages.
2	**FatalErrEn: fatal error reporting enable**. Read-write. 1=Enables sending ERR_FATAL message when a fatal error is detected.
1	**NonFatalErrEn: non-fatal error reporting enable**. Read-write. 1=Enables sending ERR_NONFATAL message when a non-fatal error is detected.
0	**CorrErrEn: correctable error reporting enable**. Read-write. 1=Enables sending ERR_CORR message when a correctable error is detected.

D1F0x64 Link Capability

Bits	Description
31:24	**PortNumber: port number**. Read-only. Value: 0. This field indicates the PCI Express® port number for the given PCI Express link.
23	Reserved.
22	**AspmOptionalityCompliance: ASP Optionality ECN capability**. Read-only. Value: 0.
21	**LinkBWNotificationCap: link bandwidth notification capability**. Read-only. Value: 0.
20	**DlActiveReportingCapable: data link layer active reporting capability**. Read-only. Value: 0.
19	**SurpriseDownErrReporting: surprise down error reporting capability**. Read-only. Value: 0.
18	**ClockPowerManagement: clock power management**. Read-only. Value: 0.
17:15	**L1ExitLatency: L1 exit latency**. Read-only. Value: 0.
14:12	**L0sExitLatency: L0s exit latency**. Read-only. Value: 0.
11:10	**PMSupport: active state power management support**. Read-only. Value: 0.

9:4	**LinkWidth: maximum link width**. Read-only. Value: 0.
3:0	**LinkSpeed: link speed**. Read-only. Value: 0.

D1F0x68 Link Control and Status

Reset: 0000_0000h.

Bits	Description
31	**LinkAutonomousBWStatus: link autonomous bandwidth status**. Read-only.
30	**LinkBWManagementStatus: link bandwidth management status**. Read-only.
29	**DlActive: data link layer link active**. Read-only. This bit indicates the status of the data link control and management state machine. Reads return a 1 to indicate the DL_Active state, otherwise 0 is returned.
28	**SlotClockCfg: slot clock configuration**. Read-only. 1=Root port uses the same clock that the platform provides.
27	**LinkTraining: link training**. Read-only. 1=Indicates that the physical layer link training state machine is in the configuration or recovery state, or that 1b was written to the RetrainLink bit but link training has not yet begun. Hardware clears this bit when the link training state machine exits the configuration/recovery state.
26	Reserved.
25:20	**NegotiatedLinkWidth: negotiated link width**. Read-only. This field indicates the negotiated width of the given PCI Express® link.
19:16	**LinkSpeed: link speed**. Read-only.
15:12	Reserved.
11	**LinkAutonomousBWIntEn: link autonomous bandwidth interrupt enable**. Read-only.
10	**LinkBWManagementEn: link bandwidth management interrupt enable**. Read-only.
9	**HWAutonomousWidthDisable: hardware autonomous width disable**. Read-only. 1=Hardware not allowed to change the link width except to correct unreliable link operation by reducing link width.
8	**ClockPowerManagementEn: clock power management enable**. Read-only.
7	**ExtendedSync: extended sync**. Read-only. 1=Forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state.
6	**CommonClockCfg: common clock configuration**. Read-only. 1=Indicates that the root port and the component at the opposite end of this Link are operating with a distributed common reference clock. 0=Indicates that the upstream port and the component at the opposite end of this Link are operating with asynchronous reference clock.
5	**RetrainLink: retrain link**. Read-only. This bit does not apply to endpoints.
4	**LinkDis: link disable**. Read-only. This bit does not apply to endpoints.
3	**ReadCplBoundary: read completion boundary**. Read-only. 0=64-byte read completion boundary.
2	Reserved.
1:0	**PmControl: active state power management enable**. Read-only. This field controls the level of ASPM supported on the given PCI Express link. Bits Definition — Bits Definition 00b Disabled. — 10b L1 Entry Enabled. 01b L0s Entry Enabled. — 11b L0s and L1 Entry Enabled.

D1F0x7C Device Capability 2

Reset: 0000_0000h.

Bits	Description
31:24	Reserved.
23:22	**MaxEndEndTlpPrefixes: Max number of End-End TLP prefixes supported**. Read-only. IF (D1F0x7C[EndEndTlpPrefixSupported] == 0) THEN Reserved. ENDIF. Bits Definition Bits Definition 00b 4 End-End TLP Prefixes. 10b 2 End-End TLP Prefixes. 01b 1 End-End TLP Prefix. 11b 3 End-End TLP Prefixes.
21	**EndEndTlpPrefixSupported: End-End TLP Prefix supported**. Read-only.
20	**ExtendedFmtFieldSupported**. Read-only. 1=Function supports 3-bit definition of PCIe® Transaction Layer Packet header FMT field. 0=Function supports 2-bit definition of FMT field.
19:18	**ObffSupported: Optimized buffer flush/fill supported**. Read-only.
17:14	Reserved.
13:12	**TphCplrSupported**. Read-only.
11	**LtrSupported: Latency Tolerance Reporting supported**. Read-only.
10	**NoRoEnabledP2pPassing**. Read-only.
9:6	Reserved.
5	**AriForwardingSupported: ARI forwarding supported**. Read-only.
4	**CplTimeoutDisSupported: completion timeout disable supported**. Read-only.
3:0	**CplTimeoutRangeSupported: completion timeout range supported**. Read-only.

D1F0x80 Device Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15	**EndEndTlpPrefixBlocking**. Read-only.
14:13	**ObffEn**. Read-only.
12:11	Reserved.
10	**LtrEn**. Read-only.
9	**IdoCompletionEn**. Read-only.
8	**IdoRequestEn**. Read-only.
7:6	Reserved.
5	**AriForwardingEn**. Read-only.
4	**CplTimeoutDis: completion timeout disable**. Read-only.
3:0	**CplTimeoutValue: completion timeout range supported**. Read-only.

D1F0x84 Link Capability 2

Bits	Description
31:9	Reserved.
8	**CrosslinkSupported: Crosslink Spported**. Read-only. Reset: 0. 1=Crosslink supported.
7:1	**SupportedLinkSpeed: Supported Link Speed**. Read-only. Reset: 07h. Specifies what link speeds are supported. Bit Definition Bit Definition [1] 2.5 GT/s [3] 8.0 GT/s [2] 5.0 GT/s [7:4] Reserved
0	Reserved.

D1F0x88 Link Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:22	Reserved.
21	**LinkEqualizationRequest**. Read-only. Set when hardware requests link equalization to be performed.
20	**EqualizationPhase3Success: Phase3 of Tx equalization procedure completed**. Read-only.
19	**EqualizationPhase2Success: Phase2 of Tx equalization procedure completed**. Read-only.
18	**EqualizationPhase1Success: Phase1 of Tx equalization procedure completed**. Read-only.
17	**EqualizationComplete: Tx equalization procedure completed**. Read-only.
16	**CurDeemphasisLevel: current deemphasis level**. Read-only. 1=-3.5 dB. 0=-6 dB.
15:13	Reserved.
12	**ComplianceDeemphasis: compliance deemphasis**. Read-only. This bit defines the deemphasis level used in compliance mode. 1=-3.5 dB. 0=-6 dB.
11	**ComplianceSOS: compliance SOS**. Read-only. 1=The device transmits skip ordered sets in between the modified compliance pattern.
10	**EnterModCompliance: enter modified compliance**. Read-only. 1=The device transmits modified compliance pattern.
9:7	**XmitMargin: transmit margin**. Read-only. This field controls the non-deemphasized voltage level at the transmitter pins.
6	**SelectableDeemphasis: selectable deemphasis**. Read-only.
5	**HwAutonomousSpeedDisable: hardware autonomous speed disable**. Read-only. 1=Disables hardware generated link speed changes.
4	**EnterCompliance: enter compliance**. Read-only. 1=Force link to enter compliance mode.
3:0	**TargetLinkSpeed: target link speed**. Read-only. This field defines the upper limit of the link operational speed.

D1F0xA0 MSI Capability

Bits	Description
31:24	Reserved.
23	**Msi64bit: MSI 64-bit capability**. Read-only. Reset: 1. 1=The device is capable of sending 64-bit MSI messages. 0=The device is not capable of sending a 64-bit message address.
22:20	**MsiMultiEn: MSI multiple message enable**. Read-write. Reset: 000b. Software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). When MSI is enabled, a function is allocated at least 1 vector.
19:17	**MsiMultiCap: MSI multiple message capability**. Read-only. Reset: 000b. 000b=The device is requesting one vector.
16	**MsiEn: MSI enable**. Read-write. Reset: 0. 1=MSI generation is enabled and INTx generation is disabled. 0=MSI generation disabled and INTx generation is enabled.
15:8	**NextPtr: next pointer**. Read-only. Reset:00h.
7:0	**CapID: capability ID**. Read-only. Reset: 05h. 05h=MSI capability structure.

D1F0xA4 MSI Message Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**MsiMsgAddrLo: MSI message address**. Read-write. This register specifies the dword aligned address for the MSI memory write transaction.
1:0	Reserved.

D1F0xA8 MSI Message Address High

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**MsiMsgAddrHi: MSI message address**. Read-write. This register specifies the upper 8 bits of the MSI address in 64-bit MSI mode.

D1F0xAC MSI Message Data

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**MsiData: MSI message data**. Read-write. This register specifies lower 16 bits of data for the MSI memory write transaction. The upper 16 bits are always 0.

D1F0x100 Vendor Specific Enhanced Capability

Reset: 0061_000Bh.

Bits	Description
31:20	**NextPtr: next pointer**. Read-only.
19:16	**CapVer: capability version**. Read-only.
15:0	**CapID: capability ID**. Read-only.

D1F0x104 Vendor Specific Header

Reset: 0101_0001h.

Bits	Description
31:20	**VsecLen: vendor specific enhanced next pointer**. Read-only.
19:16	**VsecRev: vendor specific enhanced capability version**. Read-only.
15:0	**VsecID: vendor specific enhanced capability ID**. Read-only.

D1F0x108 Vendor Specific 1

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

D1F0x10C Vendor Specific 2

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

3.6 Device 1 Function 1 (Audio Controller) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D1F1x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: Product-specific.
15:0	**VendorID: vendor ID**. Read-only. Value: 1002h.

D1F1x04 Status/Command

Reset: 0010_0000h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; Write-1-to-clear. 1=Poisoned PCIe® Transaction Layer Packet (TLP) received.
30	**SignaledSystemError: signaled system error**. Read; Write-1-to-clear. 1=A non-fatal or fatal error message was sent and SerrEn = 1.
29	**ReceivedMasterAbort: received master abort**. Read; Write-1-to-clear. 1=A completion with an unsupported request completion status was received.
28	**ReceivedTargetAbort: received target abort**. Read; Write-1-to-clear. 1=A completion with completer abort completion status was received.
27	**SignalTargetAbort: Signaled target abort**. Read-only.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**MasterDataPerr: master data parity error**. Read; Write-1-to-clear. 1=(ParityErrorEn == 1) and either a poisoned completion was received or the device poisoned a write request.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	**UDFEn: UDF enable**. Read-only.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1=Capability list supported.
19	**IntStatus: interrupt status**. Read-only. 1=INTx interrupt message pending.
18:11	Reserved.
10	**IntDis: interrupt disable**. Read-write. 1=INTx interrupt messages generation disabled.
9	**FastB2BEn: fast back-to-back enable**. Read-only.
8	**SerrEn: System error enable**. Read-write. 1=Enables reporting of non-fatal and fatal errors detected.
7	**Stepping: Stepping control**. Read-only.
6	**ParityErrorEn: parity error response enable**. Read-write.

5	**PalSnoopEn: VGA palette snoop enable**. Read-only.
4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.
2	**BusMasterEn: bus master enable**. Read-write. 1=Memory and IO read and write request generation enabled.
1	**MemAccessEn: IO access enable**. Read-write. This bit controls if memory accesses targeting this device are accepted. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-write. This bit controls if IO accesses targeting this device are accepted. 1=Enabled. 0=Disabled.

D1F1x08 Class Code/Revision ID

Reset: 0403_0000h.

Bits	Description
31:8	**ClassCode**. Read-only.
7:0	**RevID: revision ID**. Read-only.

D1F1x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:24	**BIST**. Read-only. These bits are fixed at their default values.
23:16	**HeaderTypeReg**. Read-only. 80h=Type 0 multi-function device.
15:8	**LatencyTimer**. Read-only. These bits are fixed at their default value.
7:0	**CacheLineSize**. Read-write. This field specifies the system cache line size in units of double words.

D1F1x10 Audio Registers Base Address

Reset: 0000_0000h.

Bits	Description
31:14	**BaseAddr: base address**. Read-write.
13:4	Reserved.
3	**Pref: prefetchable**. Read-only. 0=Non-prefetchable memory region.
2:1	**Type: base address register type**. Read-only. 00b=32-bit base address register.
0	**MemSpace: memory space type**. Read-only. 0=Memory mapped base address.

D1F1x14 Base Address 1

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x18 Base Address 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x1C Base Address 3

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x20 Base Address 4

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x24 Base Address 5

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x2C Subsystem and Subvendor ID

Reset: 0000_0000h. This register can be modified through D1F1x4C.

Bits	Description
31:16	**SubsystemID**. Read-only.
15:0	**SubsystemVendorID**. Read-only.

D1F1x30 Expansion ROM Base Address

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D1F1x34 Capabilities Pointer

Reset: 0000_0050h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Pointer to PM capability.

D1F1x3C Interrupt Line

Reset: 0000_02FFh.

Bits	Description
31:16	Reserved.
15:8	**InterruptPin: interrupt pin**. Read-only. This field identifies the legacy interrupt message the function uses.
7:0	**InterruptLine: interrupt line**. Read-write. This field contains the interrupt line routing information.

D1F1x4C Subsystem and Subvendor ID Mirror

Reset: 0000_0000h.

Bits	Description
31:16	**SubsystemID**. Read-write. This field sets the value in the corresponding field in D1F1x2C.
15:0	**SubsystemVendorID**. Read-write. This field sets the value in the corresponding field in D1F1x2C.

D1F1x50 Power Management Capability

Bits	Description
31:27	**PmeSupport**. Value: 0_0000b. Indicates that there is no PME support.
26	**D2Support: D2 support**. Value: 1. D2 is supported.
25	**D1Support: D1 support**. Value: 1. D1 is supported.
24:22	**AuxCurrent: auxiliary current**. Value: 0.
21	**DevSpecificInit: device specific initialization**. Value: 0. Indicates that there is no device specific initialization necessary.
20	Reserved.
19	**PmeClock**. Value: 0.
18:16	**Version: version**. Value: 011b.
15:8	**NextPtr: next pointer**. Value: 00h.
7:0	**CapID: capability ID**. Value: 01h. Indicates that the capability structure is a PCI power management data structure.

D1F1x54 Power Management Control and Status

Reset: 0000_0000h.

Bits	Description
31:24	**PmeData**. Read-only.
23	**BusPwrEn**. Read-only.
22	**B2B3Support**. Read-only. B states are not supported.
21:16	Reserved.
15	**PmeStatus: PME status**. Read-only.
14:13	**DataScale: data scale**. Read-only.
12:9	**DataSelect: data select**. Read-only.
8	**PmeEn: PME# enable**. Read-only.
7:4	Reserved.
3	**NoSoftReset: no soft reset**. Read-only. Software is required to re-initialize the function when returning from $D3_{hot}$.
2	Reserved.
1:0	**PowerState: power state**. Read-write. This 2-bit field is used both to determine the current power state of the root port and to set the root port into a new power state. Bits Definition 00b D0 10b-01b Reserved 11b $D3_{hot}$

D1F1x58 PCI Express® Capability

Bits	Description
31:30	Reserved.
29:25	**IntMessageNum: interrupt message number**. Value: 0. This field indicates which MSI vector is used for the interrupt message.
24	**SlotImplemented: Slot implemented**. Value: 0.
23:20	**DeviceType: device type**. Value: 9h.
19:16	**Version**. Value: 2h.
15:8	**NextPtr: next pointer**. Value: Product-specific.
7:0	**CapID: capability ID**. Value: 10h.

D1F1x5C Device Capability

Bits	Description
31:29	Reserved.
28	**FlrCapable: function level reset capability**. Value: Product-specific.

27:26	**CapturedSlotPowerScale: captured slot power limit scale**. Value: 0.
25:18	**CapturedSlotPowerLimit: captured slot power limit value**. Value: 0.
17:16	Reserved.
15	**RoleBasedErrReporting: role-based error reporting**. Value: 1.
14:12	Reserved.
11:9	**L1AcceptableLatency: endpoint L1 Acceptable Latency**. Value: 111b.
8:6	**L0SAcceptableLatency: endpoint L0s Acceptable Latency**. Value: 110b.
5	**ExtendedTag: extended tag support**. Value: 1. 8-bit tag support.
4:3	**PhantomFunc: phantom function support**. Value: 0. No phantom functions supported.
2:0	**MaxPayloadSupport: maximum supported payload size**. Value: 000b. 128 bytes max payload size.

D1F1x60 Device Control and Status

Reset: 0000_0810h.

Bits	Description
31:22	Reserved.
21	**TransactionsPending: transactions pending**. Read-only.
20	**AuxPwr: auxiliary power**. Read-only.
19	**UsrDetected: unsupported request detected**. Read; Write-1-to-clear. 1=Unsupported request received.
18	**FatalErr: fatal error detected**. Read; Write-1-to-clear. 1=Fatal error detected.
17	**NonFatalErr: non-fatal error detected**. Read; Write-1-to-clear. 1=Non-fatal error detected.
16	**CorrErr: correctable error detected**. Read; Write-1-to-clear. 1=Correctable error detected.
15	**BridgeCfgRetryEn: bridge configuration retry enable**. Read-only.
14:12	**MaxRequestSize: maximum request size**. Read-only. 0=The root port never generates read requests with size exceeding 128 bytes.
11	**NoSnoopEnable: enable no snoop**. Read-write. 1=The device is permitted to set the No Snoop bit in requests.
10	**AuxPowerPmEn: auxiliary power PM enable**. Read-only. This capability is not implemented.
9	**PhantomFuncEn: phantom functions enable**. Read-only. Phantom functions are not supported.
8	**ExtendedTagEn: extended tag enable**. Read-write. 1=8-bit tag request tags. 0=5-bit request tag.
7:5	**MaxPayloadSize: maximum supported payload size**. Read-only. 000b=Indicates a 128-byte maximum payload size.
4	**RelaxedOrdEn: relaxed ordering enable**. Read-write. 1=The device is permitted to set the Relaxed Ordering bit.
3	**UsrReportEn: unsupported request reporting enable**. Read-write. 1=Enables signaling unsupported requests by sending error messages.
2	**FatalErrEn: fatal error reporting enable**. Read-write. 1=Enables sending ERR_FATAL message when a fatal error is detected.

1	**NonFatalErrEn: non-fatal error reporting enable**. Read-write. 1=Enables sending ERR_NONFATAL message when a non-fatal error is detected.
0	**CorrErrEn: correctable error reporting enable**. Read-write. 1=Enables sending ERR_CORR message when a correctable error is detected.

D1F1x64 Link Capability

Bits	Description
31:24	**PortNumber: port number**. Value: 0. This field indicates the PCI Express® port number for the given PCI Express link.
23:22	Reserved.
21	**LinkBWNotificationCap: link bandwidth notification capability**. Read-only. Value: 0b.
20	**DlActiveReportingCapable: data link layer active reporting capability**. Read-only. Value: 0b.
19	**SurpriseDownErrReporting: surprise down error reporting capability**. Read-only. Value: 0b.
18	**ClockPowerManagement: clock power management**. Read-only. Value: 0b.
17:15	**L1ExitLatency: L1 exit latency**. Read-only. Value: 0b.
14:12	**L0sExitLatency: L0s exit latency**. Read-only. Value: 0b.
11:10	**PMSupport: active state power management support**. Read-only. Value: 0b.
9:4	**LinkWidth: maximum link width**. Read-only. Value: 0.
3:0	**LinkSpeed: link speed**. Read-only. Value: 0b.

D1F1x68 Link Control and Status

Reset: 0000_0000h.

Bits	Description
31	**LinkAutonomousBWStatus: link autonomous bandwidth status**. Read-only.
30	**LinkBWManagementStatus: link bandwidth management status**. Read-only.
29	**DlActive: data link layer link active**. Read-only. This bit indicates the status of the data link control and management state machine. Reads return a 1 to indicate the DL_Active state, otherwise 0 is returned.
28	**SlotClockCfg: slot clock configuration**. Read-only. 1=The root port uses the same clock that the platform provides.
27	**LinkTraining: link training**. Read-only. 1=Indicates that the physical layer link training state machine is in the configuration or recovery state, or that 1b was written to the RetrainLink bit but link training has not yet begun. Hardware clears this bit when the link training state machine exits the configuration/recovery state.
26	Reserved.
25:20	**NegotiatedLinkWidth: negotiated link width**. Read-only. This field indicates the negotiated width of the given PCI Express® link.

19:16	**LinkSpeed: link speed**. Read-only. Bits: Description 0h: Reserved 1h: 2.5 Gb/s. 2h: 5 Gb/s. Fh-3h: Reserved
15:12	Reserved.
11	**LinkAutonomousBWIntEn: link autonomous bandwidth interrupt enable**. Read-only.
10	**LinkBWManagementEn: link bandwidth management interrupt enable**. Read-only.
9	**HWAutonomousWidthDisable: hardware autonomous width disable**. Read-only. 1=Hardware not allowed to change the link width except to correct unreliable link operation by reducing link width.
8	**ClockPowerManagementEn: clock power management enable**. Read-only.
7	**ExtendedSync: extended sync**. Read-only. 1=Forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state.
6	**CommonClockCfg: common clock configuration**. Read-only. 1=Indicates that the root port and the component at the opposite end of this Link are operating with a distributed common reference clock. 0=Indicates that the upstream port and the component at the opposite end of this Link are operating with asynchronous reference clock.
5	**RetrainLink: retrain link**. Read-only. This bit does not apply to endpoints.
4	**LinkDis: link disable**. Read-only. This bit does not apply to endpoints.
3	**ReadCplBoundary: read completion boundary**. Read-only. 0=64-byte read completion boundary.
2	Reserved.
1:0	**PmControl: active state power management enable**. Read-only. This field controls the level of ASPM supported on the given PCI Express link. Bits: Definition: Bits: Definition 00b: Disabled. 10b: L1 Entry Enabled. 01b: L0s Entry Enabled. 11b: L0s and L1 Entry Enabled.

D1F1x7C Device Capability 2

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**CplTimeoutDisSup: completion timeout disable supported**. Read-only.
3:0	**CplTimeoutRangeSup: completion timeout range supported**. Read-only.

D1F1x80 Device Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**CplTimeoutDis: completion timeout disable**. Read-only.
3:0	**CplTimeoutValue: completion timeout range supported**. Read-only.

D1F1x84 Link Capability 2

Bits	Description
31:0	Reserved.

D1F1x88 Link Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:17	Reserved.
16	**CurDeemphasisLevel: current deemphasis level**. Read-only. 1=-3.5 dB. 0=-6 dB.
15:13	Reserved.
12	**ComplianceDeemphasis: compliance deemphasis**. Read-only. This bit defines the deemphasis level used in compliance mode. 1=-3.5 dB. 0=-6 dB.
11	**ComplianceSOS: compliance SOS**. Read-only. 1=The device transmits skip ordered sets in between the modified compliance pattern.
10	**EnterModCompliance: enter modified compliance**. Read-only. 1=The device transmits modified compliance pattern.
9:7	**XmitMargin: transmit margin**. Read-only. This field controls the non-deemphasized voltage level at the transmitter pins.
6	**SelectableDeemphasis: selectable deemphasis**. Read-only.
5	**HwAutonomousSpeedDisable: hardware autonomous speed disable**. Read-only. 1=Disables hardware generated link speed changes.
4	**EnterCompliance: enter compliance**. Read-only. 1=Force link to enter compliance mode.
3:0	**TargetLinkSpeed: target link speed**. Read-only. This fields defines the upper limit of the link operational speed.

D1F1xA0 MSI Capability

Bits	Description
31:24	Reserved.
23	**Msi64bit: MSI 64-bit capability**. Read-only. Reset: 1. 1=The device is capable of sending 64-bit MSI messages. 0=The device is not capable of sending a 64-bit message address.
22:20	**MsiMultiEn: MSI multiple message enable**. Read-write. Reset: 000b. Software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). When MSI is enabled, a function is allocated at least 1 vector.
19:17	**MsiMultiCap: MSI multiple message capability**. Read-only. Reset: 000b. 000b=The device is requesting one vector.
16	**MsiEn: MSI enable**. Read-write. Reset: 0. 1=MSI generation is enabled and INTx generation is disabled. 0=MSI generation disabled and INTx generation is enabled.
15:8	**NextPtr: next pointer**. Read-only. Reset: 00h.
7:0	**CapID: capability ID**. Read-only. Reset: 05h. 05h=MSI capability structure.

D1F1xA4 MSI Message Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**MsiMsgAddrLo: MSI message address**. Read-write. This register specifies the dword aligned address for the MSI memory write transaction.
1:0	Reserved.

D1F1xA8 MSI Message Address High

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:0	**MsiMsgAddrHi: MSI message address**. Read-write. This register specifies the upper 8 bits of the MSI address in 64-bit MSI mode.

D1F1xAC MSI Message Data

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**MsiData: MSI message data**. Read-write. This register specifies lower 16 bits of data for the MSI memory write transaction. The upper 16 bits are always 0.

D1F1x100 Vendor Specific Enhanced Capability

Reset: 0111_000Bh.

Bits	Description
31:20	**NextPtr: next pointer**. Read-only.
19:16	**CapVer: capability version**. Read-only.
15:0	**CapID: capability ID**. Read-only.

D1F1x104 Vendor Specific Header

Reset: 0101_0001h.

Bits	Description
31:20	**VsecLen: vendor specific enhanced next pointer**. Read-only.
19:16	**VsecRev: vendor specific enhanced capability version**. Read-only.
15:0	**VsecID: vendor specific enhanced capability ID**. Read-only.

D1F1x108 Vendor Specific 1

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

D1F1x10C Vendor Specific 2

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write.

3.7 Device 2 Function 0 (Host Bridge) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space. D2F0 registers do not control any hardware. They ensure that software can configure functions 1 through 4.

D2F0x00 Device/Vendor ID (Host Bridge)

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 157Bh.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D2F0x04 Status/Command

Reset: 0000_0000h.

Bits	Description
31:16	**Status**. Read-only.
15:0	**Command**. Read-only.

D2F0x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode: class code**. Read-only.
7:0	**RevId: revision identifier**. Read-only.

D2F0x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:24	Reserved.
23	**DeviceType**. Read-only. 1=Indicates that the northbridge block is a multi-function device. 0=Indicates that the northbridge block is a single function device.
22:16	**HeaderType**. Read-only. Indicates multiple functions present in this device.
15:0	Reserved.

D2F0x40 Header Type Write

Reset: 0000_0080h.

Bits	Description
31:8	Reserved.
7	**DeviceType**. Read-write. This field sets the value in D2F0x0C[DeviceType]. 0=Single function device. 1=Multi-function device.
6:0	Reserved.

3.8 Device 2 Function [5:1] (Root Port) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space. See 2.11.1 [Overview].

D2F[5:1]x00 Device/Vendor ID

Table 80: Register Mapping for D2F[5:1]x00

D2F[5:1]x00	Function
D2F1x00	GPP Bridge 0
D2F2x00	GPP Bridge 1
D2F3x00	GPP Bridge 2
D2F4x00	GPP Bridge 3
D2F5x00	GPP Bridge 4

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 157Ch.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D2F[5:1]x04 Status/Command Register

Reset: 0010_0000h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; Write-1-to-clear.
30	**SignaledSystemError: signaled system error**. Read; Write-1-to-clear. 1=System error signaled.
29	**ReceivedMasterAbort: received master abort**. Read; Write-1-to-clear.
28	**ReceivedTargetAbort: received target abort**. Read; Write-1-to-clear.
27	**SignalTargetAbort: signaled target abort**. Read; Write-1-to-clear.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.

24	**DataPerr: data parity error**. Read; Write-1-to-clear.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	Reserved.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only. 1=Capability list present.
19	**IntStatus: interrupt status**. Read-only. 1=An INTx interrupt Message is pending in the device.
18:11	Reserved.
10	**IntDis: interrupt disable**. Read-write.
9	**FastB2BEn: fast back-to-back enable**. Read-only.
8	**SerrEn: system error enable**. Read-write. 1=System error reporting enabled.
7	**Stepping: Stepping control**. Read-only.
6	**ParityErrorEn: parity error response enable**. Read-write.
5	**PalSnoopEn: VGA palette snoop enable**. Read-only.
4	**MemWriteInvalidateEn: memory write and invalidate enable**. Read-only.
3	**SpecialCycleEn: special cycle enable**. Read-only.
2	**BusMasterEn: bus master enable**. Read-write.
1	**MemAccessEn: IO access enable**. Read-write. This bit controls if memory accesses targeting this device are accepted or not. 1=Enabled. 0=Disabled.
0	**IoAccessEn: IO access enable**. Read-write. This bit controls if IO accesses targeting this device are accepted or not. 1=Enabled. 0=Disabled.

D2F[5:1]x08 Class Code/Revision ID Register

Reset: 0604_00XXh.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D2F[5:1]x0C Header Type Register

Reset: 0001_0000h.

Bits	Description
31:24	**BIST**. Read-only. These bits are fixed at their default values.
23	**DeviceType**. Read-only. 0=Single function device. 1=Multi-function device.
22:16	**HeaderType**. Read-only. These bits are fixed at their default values. Indicates a Type 0 or Type 1 configuration space.
15:8	**LatencyTimer**. Read-only. This field does not control any hardware.
7:0	**CacheLineSize**. Read-write.

D2F[5:1]x18 Bus Number and Secondary Latency Register

Reset: 0000_0000h.

Bits	Description
31:24	**SecondaryLatencyTimer: secondary latency timer**. Read-only. This field is always 0.
23:16	**SubBusNumber: subordinate number**. Read-write. This field contains the highest-numbered bus that exists on the secondary side of the bridge.
15:8	**SecondaryBus: secondary bus number**. Read-write. This field defines the bus number of the secondary bus interface.
7:0	**PrimaryBus: primary bus number**. Read-write. This field defines the bus number of the primary bus interface.

D2F[5:1]x1C IO Base and Secondary Status Register

Reset: 0000_0101h.

Bits	Description
31	**ParityErrorDetected: detected parity error**. Read; Write-1-to-clear. A Poisoned PCIe® Transaction Layer Packet (TLP) was received regardless of the state of the D2F[5:1]x04[ParityErrorEn].
30	**ReceivedSystemError: signaled system error**. Read; Write-1-to-clear. 1=A System Error was detected.
29	**ReceivedMasterAbort: received master abort**. Read; Write-1-to-clear. 1=A CPU transaction is terminated due to a master-abort.
28	**ReceivedTargetAbort: received target abort**. Read; Write-1-to-clear. 1=A CPU transaction (except for a special cycle) is terminated due to a target-abort.
27	**SignalTargetAbort: signaled target abort**. Read; Write-1-to-clear.
26:25	**DevselTiming: DEVSEL# Timing**. Read-only.
24	**MasterDataPerr: master data parity error**. Read; Write-1-to-clear. 1=The link received a poisoned or poisoned a downstream write and D2F[5:1]x3C[ParityResponseEn] = 1.
23	**FastBackCapable: fast back-to-back capable**. Read-only.
22	Reserved.
21	**PCI66En: 66 MHz capable**. Read-only.
20	**CapList: capability list**. Read-only.
19:16	Reserved.
15:12	**IOLimit[15:12]**. Read-write. Lower part of the limit address. Upper part is defined in D2F[5:1]x30.
11:8	**IOLimitType**. Read-only. 0=16-bit. 1=32-bit.
7:4	**IOBase[15:12]**. Read-write. Lower part of the base address. Upper part is defined in D2F[5:1]x30.
3:0	**IOBaseType**. Read-only. 0=16-bit. 1=32-bit.

D2F[5:1]x20 Memory Limit and Base Register

Reset: 0000_0000h.

Bits	Description
31:20	**MemLimit[31:20]**. Read-write.
19:16	**MemLimitType**. Read-only. 0=32-bit. 1=64-bit.
15:4	**MemBase[31:20]**. Read-write.
3:0	**MemBaseType**. Read-only. 0=32-bit. 1=64-bit.

D2F[5:1]x24 Prefetchable Memory Limit and Base Register

Reset: 0001_0001h.

Bits	Description
31:20	**PrefMemLimit**. Read-write. Lower part of the limit address. Upper part is defined in D2F[5:1]x2C.
19:16	**PrefMemLimitType**. Read-only. 0=32-bit. 1=64-bit.
15:4	**PrefMemBase[31:20]**. Read-write. Lower part of the base address. Upper part is defined in D2F[5:1]x28.
3:0	**PrefMemBaseType**. Read-only. 0=32-bit. 1=64-bit.

D2F[5:1]x28 Prefetchable Memory Base High Register

Reset: 0000_0000h.

Bits	Description
31:0	**PrefMemBase[63:32]**. Read-write. Upper part of the base address. Lower part is defined in D2F[5:1]x24.

D2F[5:1]x2C Prefetchable Memory Limit High Register

Reset: 0000_0000h.

Bits	Description
31:0	**PrefMemLimit[63:32]**. Read-write. Upper part of the limit address. Lower part is defined in D2F[5:1]x24.

D2F[5:1]x30 IO Base and Limit High Register

Reset: 0000_0000h.

Bits	Description
31:16	**IOLimit[31:16]**. Read-write. Upper part of the limit address. Lower part is defined in D2F[5:1]x1C.
15:0	**IOBase[31:16]**. Read-write. Upper part of the base address. Lower part is defined in D2F[5:1]x1C.

D2F[5:1]x34 Capabilities Pointer Register

Reset: 0000_0050h.

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Pointer to PM capability.

D2F[5:1]x3C Bridge Control Register

Reset: 0000_00FFh.

Bits	Description
31:24	Reserved.
23	**FastB2BCap: Fast back-to-back capability**. Read-only.
22	**SecondaryBusReset: Secondary bus reset**. Read-write. Setting this bit triggers a hot reset on the corresponding PCI Express® Port.
21	**MasterAbortMode: Master abort mode**. Read-only.
20	**Vga16En: VGA IO 16 bit decoding enable**. Read-write. 1=Address bits[15:10] for VGA IO cycles are decoded. 0=Address bits[15:10] for VGA IO cycles are ignored.
19	**VgaEn: VGA enable**. Read-write. Affects the response by the bridge to compatible VGA addresses. When it is set, the bridge decodes and forwards the following accesses on the primary interface to the secondary interface: Memory accesses in the range of A_0000h to B_FFFFh and IO address where address bits[9:0] are in the ranges of 3B0h to 3BBh or 3C0h to 3DFh. For IO cycles the decoding of address bits[15:10] depends on Vga16En.
18	**IsaEn: ISA enable**. Read-write.
17	**SerrEn: SERR enable**. Read-write.
16	**ParityResponseEn: Parity response enable**. Read-write. Controls the bridge's response to poisoned TLPs on its secondary interface. 1=The bridge takes its normal action when a poisoned TLP is received. 0=The bridge ignores any poisoned TLPs that it receives and continues normal operation.
15:11	**IntPinR: interrupt pin**. Read-only.
10:8	**IntPin: interrupt pin**. IF (D0F0xE4_x0140_0010[HwInitWrLock] == 1) THEN Read-only. ELSE Read-write. ENDIF.
7:0	**IntLine: Interrupt line**. Read-write.

D2F[5:1]x50 Power Management Capability Register

Reset: 0003_5801h.

Bits	Description
31:27	**PmeSupport**. Read-only.
26	**D2Support: D2 support**. Read-only. D2 is not supported.
25	**D1Support: D1 support**. Read-only. D1 is not supported.
24:22	**AuxCurrent: auxiliary current**. Read-only. Auxiliary current is not supported.

21	**DevSpecificInit: device specific initialization**. Read-only. This field is hardwired to 0 to indicate that there is no device specific initialization necessary.
20	Reserved.
19	**PmeClock**. Read-only. 0=Indicate that PCI clock is not needed to generate PME messages.
18:16	**Version: version**. Read-only. 3=PMI Spec 1.2.
15:8	**NextPtr: next pointer**. Read-only. 58h=Address of the next capability structure.
7:0	**CapID: capability ID**. Read-only. 01h=PCI power management data structure.

D2F[5:1]x54 Power Management Control and Status Register

Bits	Description
31:24	**PmiData**. Read-only. Reset: 0.
23	**BusPwrEn**. Read-only. Reset: 0.
22	**B2B3Support**. Read-only. Reset: 0. B-states are not supported.
21:16	Reserved.
15	**PmeStatus: PME status**. Read; Write-1-to-clear. Reset: 0. This bit is set when the root port would issue a PME message (independent of the state of the PmeEn bit). Once set, this bit remains set until it is reset by writing a 1 to this bit location. Writing a 0 has no effect.
14:13	**DataScale: data scale**. Read-only. Reset: 0.
12:9	**DataSelect: data select**. Read-only. Reset: 0.
8	**PmeEn: PME# enable**. Read-write. Reset: 0.
7:4	Reserved.
3	**NoSoftReset: no soft reset**. Read-only. Reset: 0. Software is required to re-initialize the function when returning from $D3_{hot}$.
2	Reserved.
1:0	**PowerState: power state**. Read-write. Reset: 0. This 2-bit field is used both to determine the current power state of the root port and to set the root port into a new power state. Bits Definition Bits Definition 00b D0 10b Reserved 01b Reserved 11b D3

D2F[5:1]x58 PCI Express® Capability Register

Reset: 0042_A010h.

Bits	Description
31:30	Reserved.
29:25	**IntMessageNum: interrupt message number**. Read-only. This register indicates which MSI vector is used for the interrupt message.
24	**SlotImplemented: Slot implemented**. Read-only. 1=The IO Link associated with this port is connected to a slot.
23:20	**DeviceType: device type**. Read-only. 4h=Root complex.

19:16	**Version**. Read-only. 2h=GEN 2 compliant.
15:8	**NextPtr: next pointer**. Read-only. A0h=Pointer to the next capability structure.
7:0	**CapID: capability ID**. Read-only. 10h=PCIe® Capability structure.

D2F[5:1]x5C Device Capability Register

Reset: 1000_0020h.

Bits	Description
31:29	Reserved.
28	**FlrCapable: function level reset capability**. Read-only.
27:26	**CapturedSlotPowerScale: captured slot power limit scale**. Read-only.
25:18	**CapturedSlotPowerLimit: captured slot power limit value**. Read-only.
17:16	Reserved.
15	**RoleBasedErrReporting: role-based error reporting**. Read-only.
14:12	Reserved.
11:9	**L1AcceptableLatency: endpoint L1 Acceptable Latency**. Read-only.
8:6	**L0SAcceptableLatency: endpoint L0s Acceptable Latency**. Read-only.
5	**ExtendedTag: extended tag support**. Read-only. 1=8-bit tag supported. 0=5-bit tag supported.
4:3	**PhantomFunc: phantom function support**. Read-only. 0=No phantom functions supported.
2:0	**MaxPayloadSupport: maximum supported payload size**. Read-only. 000b=128 bytes max payload size.

D2F[5:1]x60 Device Control and Status Register

Reset: 0000_0000h.

Bits	Description
31:22	Reserved.
21	**TransactionsPending: transactions pending**. Read-only. 0=No internally generated non-posted transactions pending.
20	**AuxPwr: auxiliary power**. Read-only.
19	**UsrDetected: unsupported request detected**. Read; Write-1-to-clear. 1=The port received an unsupported request. Errors are logged in this register even if error reporting is disabled.
18	**FatalErr: fatal error detected**. Read; Write-1-to-clear. 1=The port detected a fatal error. Errors are logged in this register even if error reporting is disabled.
17	**NonFatalErr: non-fatal error detected**. Read; Write-1-to-clear. 1=The port detected a non-fatal error. Errors are logged in this register even if error reporting is disabled.
16	**CorrErr: correctable error detected**. Read; Write-1-to-clear. 1=The port detected a correctable error. Errors are logged in this register even if error reporting is disabled.
15	**InitiateFlr**. Read-write. 1=Enable function level reset.
14:12	**MaxRequestSize: maximum request size**. Read-write.

11	**NoSnoopEnable: enable no snoop**. Read-write. 1=The port is permitted to set the No Snoop bit in the Requester Attributes of transactions it initiates that do not require hardware enforced cache coherency.
10	**AuxPowerPmEn: auxiliary power PM enable**. Read-only.
9	**PhantomFuncEn: phantom functions enable**. Read-only.
8	**ExtendedTagEn: extended tag enable**. Read-write. 1=8-bit tag generations enabled. 0=5-bit tags are used.
7:5	**MaxPayloadSize: maximum supported payload size**. Read-write. Bits Definition Bits Definition 0h 128B 3h 1024B 1h 256B 4h 2048B 2h 512B 5h 4096B
4	**RelaxedOrdEn: relaxed ordering enable**. Read-write. 1=The root port is permitted to set the relaxed ordering bit in the attributes field of transactions it initiates that do not require strong write ordering.
3	**UsrReportEn: unsupported request reporting enable**. Read-write. 1=Reporting of unsupported requests enabled.
2	**FatalErrEn: fatal error reporting enable**. Read-write. 1=Enable sending ERR_FATAL messages.
1	**NonFatalErrEn: non-fatal error reporting enable**. Read-write. 1=Enable sending ERR_NONFATAL messages.
0	**CorrErrEn: correctable error reporting enable**. Read-write. 1=Enable sending ERR_CORR messages.

D2F[5:1]x64 IO Link Capability Register

Read-only.

Bits	Description
31:24	**PortNumber: port number**. Reset: 0. This field indicates the port number for the given IO link.
23	Reserved.
22	**AspmOptionalityCompliance**. Reset: 1. This field indicates if the compenent supports the ASPM Optionality ECN.
21	**LinkBWNotificationCap: link bandwidth notification capability**. Reset: 0.
20	**DlActiveReportingCapable: data link layer active reporting capability**. Reset: 0.
19	**SurpriseDownErrReporting**. Reset: 0. 1=This field indicates if the component supports the detecting and reporting of a Surprise Down error condition.
18	**ClockPowerManagement: clock power management**. Reset: 0. 0=Indicates that the reference clock must not be removed while in L1 or L2/L3 ready link states.
17:15	**L1ExitLatency: L1 exit latency**. Reset: 010b. 010b=Indicate an exit latency between 2 us and 4 us.
14:12	**L0sExitLatency: L0s exit latency**. Reset: 001b. 001b=Indicates an exit latency between 64 ns and 128 ns.
11:10	**PMSupport: active state power management support**. Reset: 11b. 11b=Indicates support of L0s and L1.

9:4	**LinkWidth: maximum link width**. Value: 10h. Bits — Definition 00h — Reserved. 01h — 1 lane. 02h — 2 lanes. 03h — Reserved. 04h — 4 lanes. 07h-05h — Reserved. 08h — 8 lanes. 0Bh-09h — Reserved. 0Ch — 12 lanes. 0Fh-0Dh — Reserved. 10h — 16 lanes. 1Fh-11h — Reserved. 20h — 32 lanes. 3Fh-21h — Reserved.
3:0	**LinkSpeed: link speed**. Value: IF (D2F[5:1]xE4_xA4[LcGen2EnStrap] == 0 && D2F[5:1]xE4_xA4[LcGen3EnStrap] == 0) THEN 1h. ELSEIF (D2F[5:1]xE4_xA4[LcGen2EnStrap] == 1 && D2F[5:1]xE4_xA4[LcGen3EnStrap] == 0) THEN 2h. ELSEIF (D2F[5:1]xE4_xA4[LcGen2EnStrap] == 0 && D2F[5:1]xE4_xA4[LcGen3EnStrap] == 1) THEN 3h. ELSEIF (D2F[5:1]xE4_xA4[LcGen2EnStrap] == 1 && D2F[5:1]xE4_xA4[LcGen3EnStrap] == 1) THEN 3h. ENDIF. Bits — Definition 0h — Reserved. 1h — 2.5 Gb/s. 2h — 5.0 Gb/s. 3h — 8.0 Gb/s. Fh-4h — Reserved.

D2F[5:1]x68 IO Link Control and Status Register

Reset: 1001_0000h.

Bits	Description
31	**LinkAutonomousBWStatus: link autonomous bandwidth status**. IF (D2F[5:1]x64[LinkBWNotificationCap] == 0) THEN Read-only. ELSE Read-write; Updated-by-hardware. ENDIF.
30	**LinkBWManagementStatus: link bandwidth management status**. IF (D2F[5:1]x64[LinkBWNotificationCap] == 0) THEN Read-only. ELSE Read-write; Updated-by-hardware. ENDIF.
29	**DlActive: data link layer link active**. Read-only; Updated-by-hardware. This bit indicates the status of the data link control and management state machine. 1=DL_Active state. 0=All other states.
28	**SlotClockCfg: slot clock configuration**. Read-only; Updated-by-hardware. 1=The root port uses the same clock that the platform provides.
27	**LinkTraining: link training**. Read-only; Updated-by-hardware. This bit indicates that the physical layer link training state machine is in the configuration or recovery state, or that 1b was written to the RetrainLink bit but link training has not yet begun. Hardware clears this bit when the link training state machine exits the configuration/recovery state.

26	Reserved.
25:20	**NegotiatedLinkWidth: negotiated link width**. Read-only; Updated-by-hardware. This field indicates the negotiated width of the given PCI Express® link. Bits — Definition 00h — Reserved. 01h — 1 lane. 02h — 2 lanes. 03h — Reserved. 04h — 4 lanes. 07h-05h — Reserved. 08h — 8 lanes. 0Bh-09h — Reserved. 0Ch — 12 lanes. 0Fh-0Dh — Reserved. 10h — 16 lanes. 1Fh-11h — Reserved. 20h — 32 lanes. 3Fh-21h — Reserved.
19:16	**LinkSpeed: link speed**. Read-only; Updated-by-hardware. Bits — Definition 00h — Reserved. 01h — 2.5 Gb/s. 02h — 5.0 Gb/s. Fh-3h — Reserved.
15:12	Reserved.
11	**LinkAutonomousBWIntEn: link autonomous bandwidth interrupt enable**. Read-write. 1=Enables the generation of an interrupt to indicate that the Link AutonomousBWStatus bit has been set.
10	**LinkBWManagementIntEn: link bandwidth management interrupt enable**. Read-write. 1=Enables the generation of an interrupt to indicate that the LinkBWManagementStatus has been set.
9	**HWAutonomousWidthDisable: hardware autonomous width disable**. Read-write. 1=Disables hardware from changing the link width for reasons other than attempting to correct unreliable link operation by reducing link width.
8	**ClockPowerManagementEn: clock power management enable**. Read-write.
7	**ExtendedSync: extended sync**. Read-write. 1=Forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state.
6	**CommonClockCfg: common clock configuration**. Read-write. 1=Indicates that the root port and the component at the opposite end of this IO link are operating with a distributed common reference clock. 0=Indicates that the root port and the component at the opposite end of this IO Link are operating with asynchronous reference clock.
5	**RetrainLink: retrain link**. Read-write; Cleared-when-done. 1=Initiate link retraining.
4	**LinkDis: link disable**. Read-write. 1=Disable link. Writes to this bit are immediately reflected in the value read from the bit, regardless of actual link state.
3	**ReadCplBoundary: read completion boundary**. Read-only. 0=64-byte read completion boundary.

2	Reserved.
1:0	**PmControl: active state power management enable**. Read-write. This field controls the level of ASPM supported on the given IO link. Bits Definition Bits Definition 00b Disabled. 10b L1 Entry Enabled. 01b L0s Entry Enabled. 11b L0s and L1 Entry Enabled.

D2F[5:1]x6C Slot Capability Register

Reset: 0004_0000h.

Bits	Description
31:19	**PhysicalSlotNumber: physical slot number**. Read-write. This field indicates the physical slot number attached to this port. This field is set to a value that assigns a slot number that is unique within the chassis, regardless of the form factor associated with the slot. This field must be initialized to 0 for ports connected to devices that are on the system board.
18	**NoCmdCplSupport: no command completed support**. Read-write. 1=Indicates that this slot does not generate software notification when an issued command is completed by the hot-plug controller.
17	**ElecMechIlPresent: electromechanical interlock present**. Read-write. 0=Indicates that a electromechanical interlock is not implemented for this slot.
16:15	**SlotPwrLimitScale: slot power limit scale**. Read-write. Specifies the scale used for the SlotPwrLimitValue. Range of Values: Bits Definition Bits Definition 00b 1.0 10b 0.01 01b 0.1 11b 0.001
14:7	**SlotPwrLimitValue: slot power limit value**. Read-write. In combination with the SlotPwrLimitScale value, specifies the upper limit on power supplied by slot. Power limit (in Watts) calculated by multiplying the value in this field by the value in the SlotPwrLimitScale field.
6	**HotplugCapable: hot-plug capability**. Read-write. 1=Indicates that this slot is capable of supporting hot-plug operations.
5	**HotplugSurprise: hot-plug surprise**. Read-write. 1=Indicates that an adapter present in this slot might be removed from the system without any prior notification.
4	**PwrIndicatorPresent: power indicator present**. Read-write. 0=Indicates that a power indicator is not implemented for this slot.
3	**AttnIndicatorPresent: attention indicator present**. Read-write. 0=Indicates that a attention indicator is not implemented for this slot.
2	**MrlSensorPresent: manual retention latch sensor present**. Read-write. 0=Indicates that a manual retention latch sensor is not implemented for this slot.
1	**PwrControllerPresent: power controller present**. Read-write. 0=A power controller is not implemented for this slot.
0	**AttnButtonPresent: attention button present**. Read-write. 0=An attention button is not implemented for this slot.

D2F[5:1]x70 Slot Control and Status Register

IF (D2F[5:1]x58[SlotImplemented] == 0) THEN Reset: 0040_0000h. ELSE Reset: 0000_0000h. ENDIF.

Bits	Description
31:25	Reserved.
24	**DlStateChanged: data link layer state change**. Read; Write-1-to-clear. This bit is set when the value reported in the D2F[5:1]x68[DlActive] is changed. In response to a data link layer state changed event, software must read D2F[5:1]x68[DlActive] to determine if the link is active before initiating configuration cycles to the hot plugged device.
23	**ElecMechIlSts: electromechanical interlock status**. Read-only.
22	**PresenceDetectState: presence detect state**. Read-only. This bit indicates the presence of an adapter in the slot based on the physical layer in-band presence detect mechanism. The in-band presence detect mechanism requires that power be applied to an adapter for its presence to be detected. 0=Slot empty. 1=Card present in slot. For root ports not connected to slots (D2F[5:1]x58[SlotImplemented] == 0b), this bit always returns 1.
21	**MrlSensorState**. Read-only.
20	**CmdCpl: command completed**. Read-only.
19	**PresenceDetectChanged: presence detect changes**. Read; Write-1-to-clear. This bit is set when the value reported in PresenceDetectState is changed.
18	**MrlSensorChanged**. Read; Write-1-to-clear.
17	**PwrFaultDetected**. Read; Write-1-to-clear.
16	**AttnButtonPressed: attention button pressed**. Read-only.
15:13	Reserved.
12	**DlStateChangedEn: data link layer state changed enable**. Read-write. 1=Enables software notification when D2F[5:1]x68[DlActive] is changed.
11	**ElecMechIlCntl: electromechanical interlock control**. Read-only.
10	**PwrControllerCntl: power controller control**. Read-only.
9:8	**PwrIndicatorCntl: power indicator control**. Read-only.
7:6	**AttnIndicatorControl: attention indicator control**. Read-only.
5	**HotplugIntrEn: hot-plug interrupt enable**. Read-only.
4	**CmdCplIntrEn: command complete interrupt enable**. Read-only.
3	**PresenceDetectChangedEn: presence detect changed enable**. Read-only.
2	**MrlSensorChangedEn: manual retention latch sensor changed enable**. Read-only.
1	**PwrFaultDetectedEn: power fault detected enable**. Read-only.
0	**AttnButtonPressedEn: attention button pressed enable**. Read-only.

D2F[5:1]x74 Root Complex Capability and Control Register

Reset: 0001_0000h.

Bits	Description
31:17	Reserved.
16	**CrsSoftVisibility: CRS software visibility**. Read-only. 1=Indicates that the root port supports returning configuration request retry status (CRS) completion status to software.
15:5	Reserved.
4	**CrsSoftVisibilityEn: CRS software visibility enable**. Read-write. 1=Enables the root port returning configuration request retry status (CRS) completion status to software.
3	**PmIntEn: PME interrupt enable**. Read-write. 1=Enables interrupt generation upon receipt of a PME message as reflected D2F[5:1]x78[PmeStatus]. A PME interrupt is also generated if D2F[5:1]x78[PmeStatus] = 1 and this bit is set by software.
2	**SerrOnFatalErrEn: system error on fatal error enable**. Read-write. 1=Indicates that a system error should be generated if a fatal error (ERR_FATAL) is reported by any of the devices in the hierarchy associated with this root port, or by the root port itself.
1	**SerrOnNonFatalErrEn: system error on non-fatal error enable**. Read-write. 1=Indicates that a system error should be generated if a non-fatal error (ERR_NONFATAL) is reported by any of the devices in the hierarchy associated with this root port, or by the root port itself.
0	**SerrOnCorrErrEn: system error on correctable error enable**. Read-write. 1=Indicates that a system error should be generated if a correctable error (ERR_COR) is reported by any of the devices in the hierarchy associated with this root port, or by the root port itself.

D2F[5:1]x78 Root Complex Status Register

Reset: 0000_0000h.

Bits	Description
31:18	Reserved.
17	**PmePending: PME pending**. Read-only. This bit indicates that another PME is pending when PmeStatus is set. When PmeStatus is cleared by software; the PME is delivered by hardware by setting the PmeStatus bit again and updating the requestor ID appropriately. PmePending is cleared by hardware if no more PMEs are pending.
16	**PmeStatus: pme status**. Read; Write-1-to-clear. 1=PME was asserted by the requestor ID indicated in the PmeRequestorID field. Subsequent PMEs are kept pending until PmeStatus is cleared by writing a 1.
15:0	**PmeRequestorId: pme requestor ID**. Read-only. This field indicates the PCI requestor ID of the last PME requestor.

D2F[5:1]x7C Device Capability 2

Reset: 0000_0000h.

Bits	Description
31:24	Reserved.

23:22	**MaxEndEndTlpPrefixes: Max number of End-End TLP prefixes supported**. Read-only. IF (D2F[5:1]x7C[EndEndTlpPrefixSupported] == 0) THEN Reserved. ENDIF. Bits Definition Bits Definition 00b 4 End-End TLP Prefixes. 10b 2 End-End TLP Prefixes. 01b 1 End-End TLP Prefix. 11b 3 End-End TLP Prefixes.
21	**EndEndTlpPrefixSupported: End-End TLP Prefix supported**. Read-only.
20	**ExtendedFmtFieldSupported**. Read-only. 1=Supports the 3-bit definition of the PCIe® Transaction Layer Packet header FMT field. 0=Supports the 2-bit definition of the FMT field. BIOS: Must be set for functions that support End-End TLP prefixes.
19:18	**ObffSupported: optimized buffer flush/fill supported**. Read-only.
17:14	Reserved.
13:12	**TphCplrSupported**. Read-only.
11	**LtrSupported: latency tolerance supported**. Read-only.
10	**NoRoEnabledP2pPassing**. Read-only. 1=When set, the routing element never carries out the passing permitted that is associated with the Relaxed Ordering attribute field being set.
9	**Cas128CmpltSupported**. Read-only. Atomic 128-bit CAS as completer supported.
8	**Atomicop64CmpltSupported.** Read-only. Atomic 64-bit completer supported.
7	**Atomicop32CmpltSupported**. Read-only. Atomic 32-bit completer supported.
6	**AtomicopRoutingSupported.** Read-only. Atomic routing supported.
5	**AriForwardingSupported**. Read-only.
4	**CplTimeoutDisSupported: completion timeout disable supported**. Read-only.
3:0	**CplTimeoutRangeSupported: completion timeout range supported**. Read-only. Fh=Completion timeout range is 64s to 50us.

D2F[5:1]x80 Device Control and Status 2

Reset: 0000_8000h.

Bits	Description
31:16	Reserved.
15	**EndEndeTlpPrefixBlocking**. Read-only. 1=Forwarding of End-End PCIe® TLP Prefixes is not supported. This bit is hardwired to 1.
14:13	**ObffEn: optimized buffer flush/fill enable**. Read-write.
12:11	Reserved.
10	**LtrEn: latency tolerance reporting enable**. Read-write.
9	**IdoCompletionEnable**. Read-write. 1=Enable to set PCIe ID-Based Ordering (IDO) bit on returned completions.
8	**IdoRequestEnable**. Read-write. 1=Enable to set IDO bit on initiated requests.
7	**AtomicopEgressBlocking**. Read-write. Blocking atomics that target going out of this Egress Port.

6	**AtomicopRequestEn**. Read-write. Enable for sending Master Atomics.
5	**AriForwardingEn**. Read-write.
4	**CplTimeoutDis: completion timeout disable**. Read-write. 1=Completion timeout disabled.
3:0	**CplTimeoutValue: completion timeout value**. Read-write. BIOS: 6h. Bits Timeout Range Bits Timeout Range 0h 50ms-50us 9h 900ms-260ms 1h 100us-50us Ah 3.5s-1s 2h 10ms-1ms Ch-Bh Reserved 4h-3h Reserved Dh 13s-4s 5h 55ms-16ms Eh 64s-4s 6h 210ms-65ms Fh Reserved 8h-7h Reserved

D2F[5:1]x84 IO Link Capability 2

Bits	Description
31:9	Reserved.
8	**CrossLinkSupported**. Read-only. Reset: 0.
7:1	**SupportedLinkSpeed**. Read-only. Reset: 7h. Specifies the supported link speeds. Bit Definition Bit Definition [1] 2.5 GT/s [3] 8.0 GT/s [2] 5.0 GT/s [7:4] Reserved
0	Reserved.

D2F[5:1]x88 IO Link Control and Status 2

Bits	Description
31:22	Reserved.
21	**LinkEqualizationRequest**. Read; Write-1-to-clear. Reset: 0. 1=Hardware requests link equalization to be performed.
20	**EqualizationPhase3Success**. Read; Write-1-to-clear. Reset: 0. 1=Phase 3 of the Transmitter Equalization procedure has completed successfully.
19	**EqualizationPhase2Success**. Read; Write-1-to-clear. Reset: 0. 1=Phase 2 of the Transmitter Equalization procedure has completed successfully.
18	**EqualizationPhase1Success**. Read; Write-1-to-clear. Reset: 0. 1=Phase 1 of the Transmitter Equalization procedure has completed successfully.
17	**EqualizationComplete**. Read; Write-1-to-clear. Reset: 0. 1=Transmitter Equalization procedure has completed.
16	**CurDeemphasisLevel: current deemphasis level**. Read-only. Reset: D2F[5:1]xE4_xA4[LcGen2EnStrap]. 1=-3.5 dB. 0=-6 dB

15:12	**ComplianceDeemphasis: compliance deemphasis**. Read-write. Reset: 0. In Gen2, this field defines the compliance deemphasis level when EnterCompliance is set. Software should leave this field in its default state. Bits — Definition 0h — DeEmph=-6 dB, Preshoot=0 dB. 1h — DeEmph=-3.5 dB, Preshoot=0 dB. 2h — DeEmph=-4.5 dB, Preshoot=0 dB. 3h — DeEmph=-2.5 dB, Preshoot=0 dB. 4h — DeEmph=-0 dB, Preshoot=0 dB. 5h — DeEmph=-0 dB, Preshoot=2 dB. 6h — DeEmph=-0 dB, Preshoot=2.5 dB. 7h — DeEmph=-6 dB, Preshoot=3.5 dB. 8h — DeEmph=-3.5 dB, Preshoot=3.5 dB. 9h — DeEmph=-0 dB, Preshoot=3.5 dB. Fh-Ah — Reserved.
11	**ComplianceSOS: compliance SOS**. Read-write. Reset: 0. 1=The device transmits skip ordered sets in between the modified compliance pattern.
10	**EnterModCompliance: enter modified compliance**. Read-write. Reset: 0. 1=The device transmits modified compliance pattern. Software should leave this field in its default state.
9:7	**XmitMargin: transmit margin**. Read-write. Reset: 0. This field controls the non-deemphasized voltage level at the transmitter pins. Software should leave this field in its default state.
6	**SelectableDeemphasis: selectable deemphasis**. Read-only. Reset: D2F[5:1]xE4_xA4[LcGen2EnStrap]. 0=Selectable deemphasis is not supported. 1=Selectable deemphasis supported.
5	**HwAutonomousSpeedDisable: hardware autonomous speed disable**. Read-write. Cold reset: 0. 1=Support for hardware changing the link speed for device specific reasons disabled.
4	**EnterCompliance: enter compliance**. Read-write. Cold reset: 0. 1=Force the link to enter the compliance mode.
3:0	**TargetLinkSpeed: target link speed**. Read-write. Reset: 2h. This field defines the upper limit of the link operational speed. Writes of reserved encodings are not valid. Hardware prevents writes of reserved encodings from changing the state of this field. Bits — Definition 0h — Reserved 1h — 2.5GT/s 2h — 5.0GT/s Fh-3h — Reserved

D2F[5:1]x8C Slot Capability 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D2F[5:1]x90 Slot Control and Status 2

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D2F[5:1]xA0 MSI Capability Register

Reset: 0000_C005h.

Bits	Description
31:25	Reserved.
24	**MsiPervectorMaskingCap.** Read-only. Specifies if function supports per-vector masking. 0=Function does not support MSI per-vector masking. 1=Function supports MSI per-vector masking.
23	**Msi64bit: MSI 64-bit capability**. Read-only. 1=The device is capable of sending 64-bit MSI messages. 0=The device is not capable of sending a 64-bit message address.
22:20	**MsiMultiEn: MSI multiple message enable**. Read-write. Software writes to this field to indicate the number of allocated vectors (equal to or less than the number of requested vectors). When MSI is enabled, a function is allocated at least 1 vector.
19:17	**MsiMultiCap: MSI multiple message capability**. Read-only. 000b=The device is requesting one vector.
16	**MsiEn: MSI enable**. Read-write. 1=MSI generation is enabled and INTx generation is disabled. 0=MSI generation disabled and INTx generation is enabled.
15:8	**NextPtr: next pointer**. Read-only.
7:0	**CapID: capability ID**. Read-only. 05h=MSI capability structure.

D2F[5:1]xA4 MSI Message Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**MsiMsgAddrLo: MSI message address**. Read-write. This register specifies the dword aligned address for the MSI memory write transaction.
1:0	Reserved.

D2F[5:1]xA8 MSI Message Address High

Reset: 0000_0000h.

Bits	Description
31:0	**MsiMsgAddrHi: MSI message address**. Read-write. This register specifies the upper 32 bits of the MSI address.

D2F[5:1]xAC MSI Message Data

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**MsiData: MSI message data**. Read-write. This register specifies lower 16 bits of data for the MSI memory write transaction. The upper 16 bits are always 0.

D2F[5:1]xC0 Subsystem and Subvendor Capability ID Register

Reset: 0000_C80Dh.

Bits	Description
31:16	Reserved.
15:8	**NextPtr: next pointer**. Read-only.
7:0	**CapID: capability ID**. Read-only.

D2F[5:1]xC4 Subsystem and Subvendor ID Register

Reset: 0000_0000h.

Bits	Description
31:16	**SubsystemID**. Read-only.
15:0	**SubsystemVendorID**. Read-only.

D2F[5:1]xC8 MSI Mapping Capability

Reset: A803_0008h.

Bits	Description
31:27	**CapType: capability type**. Read-only.
26:18	Reserved.
17	**FixD**. Read-only.
16	**En**. Read-only.
15:8	**NextPtr: next pointer**. Read-only.
7:0	**CapID: capability ID**. Read-only.

D2F[5:1]xCC MSI Mapping Address Low

Bits	Description
31:20	**MsiMapAddrLo**. Read-only. Reset: 0. Lower 32 bits of the MSI address.
19:0	Reserved.

D2F[5:1]xD0 MSI Mapping Address High

Bits	Description
31:0	**MsiMapAddrHi**. Read-only. Reset: 0. Upper 32 bits of the MSI address.

D2F[5:1]xE0 Root Port Index

Reset: 0000_0000h.

The index/data pair registers, D2F[5:1]xE0 and D2F[5:1]xE4, are used to access the registers at D2F[5:1]xE4_x[FF:00]. To access any of these registers, the address is first written into the index register, D2F[5:1]xE0, and then the data is read from or written to the data register, D2F[5:1]xE4.

Bits	Description
31:8	Reserved.
7:0	**PcieIndex**. Read-write.

D2F[5:1]xE4 Root Port Data

See D2F[5:1]xE0. Address: D2F[5:1]xE0[PcieIndex].

Bits	Description
31:0	**PcieData**. Read-write.

D2F[5:1]xE4_x20 Root Port TX Control

Reset: 0050_8000h.

Bits	Description
31:16	Reserved.
15	**TxFlushTlpDis: TLP flush disable**. Read-write. 1=Disable flushing TLPs when the link is down.
14:0	Reserved.

D2F[5:1]xE4_x50 Root Port Lane Status

Reset: 0000_0000h.

Bits	Description
31:7	Reserved.
6:1	**PhyLinkWidth: port link width**. Read-only; Updated-by-hardware. Bits / Definition / Bits / Definition 00_0000b disabled 00_1000b x8 00_0001b x1 01_0000b x12 00_0010b x2 10_0000b x16 00_0100b x4
0	**PortLaneReversal: port lane reversal**. Read-only. 1=Port lanes order is reversed.

D2F[5:1]xE4_x6A Root Port Error Control

Reset: 0000_0500h.

Bits	Description
31:1	Reserved.
0	**ErrReportingDis: advanced error reporting disable**. Read-write. BIOS: 0. 1=Error reporting disabled. 0=Error reporting enabled.

D2F[5:1]xE4_x70 Root Port Receiver Control

Reset: 0108_4000h.

Bits	Description
31:20	Reserved.
19	**RxRcbCplTimeoutMode: RCB completion timeout mode**. Read-write. BIOS: 1. 1=Timeout on link down.
18:16	**RxRcbCplTimeout: RCB completion timeout**. Read-write. Bits Definition Bits Definition 000b Disabled 100b 50ms 001b 50us 101b 100ms 010b 10ms 110b 500ms 011b 25ms 111b 1ms
15:0	Reserved.

D2F[5:1]xE4_xA0 Per Port Link Controller (LC) Control

Reset: 4000_0000h.

Bits	Description
31:24	Reserved.
23	**LcL1ImmediateAck: immediate ACK ASPM L1 entry**. Read-write. BIOS: 1. 1=Always ACK ASPM L1 entry DLLPs.
22:16	Reserved.
15:12	**LcL1Inactivity: L1 inactivity timer**. Read-write. Bits Definition Bits Definition 0h L1 disabled 8h 400us 1h 1us 9h 1ms 2h 2us Ah 40us 3h 4us Bh 10ms 4h 10us Ch 40ms 5h 20us Dh 100ms 6h 40us Eh 400ms 7h 100us Fh Reserved

11:8	**LcL0sInactivity: L0s inactivity timer**. Read-write. Bits Definition Bits Definition 0h L0s disabled 8h 4us 1h 40ns 9h 10us 2h 80ns Ah 40us 3h 120ns Bh 100us 4h 200ns Ch 400us 5h 400ns Dh 1ms 6h 1us Eh 4ms 7h 2us Fh Reserved
7:4	**Lc16xClearTxPipe**. Read-write. BIOS: 3h. Specifies the number of clock to drain the TX pipe.
3:0	Reserved.

D2F[5:1]xE4_xA1 LC Training Control

Reset: B400_9880h.

Bits	Description
31:12	Reserved.
11	**LcDontGotoL0sifL1Armed: prevent Ls0 entry is L1 request in progress**. Read-write. BIOS: 1. 1=Prevent the LTSSM from transitioning to Rcv_L0s if an acknowledged request to enter L1 is in progress.
10:0	Reserved.

D2F[5:1]xE4_xA2 LC Link Width Control

Reset: 1AA0_0006h.

Bits	Description
31:30	Reserved.
29	**LcBypassRxL0sOnShortEi.** Read-write. Avoid unnecessary entry into RxL0s when receivers have done a quick entry and exit into and out of EI. 0=Always enter RxL0s on EIOS detection. 1=Bypass RxL0s Entry if EI exit detected after EIOS but before entry into RxL0s.
28	**LcHoldEiForRspeedCmdChange.** Read-write. Extends received EI until PHY command change is done for the Transmitter. 0=Don't remember that iANY_ENTER_eIDLE was received in Recovery.Speed. 1=Hold received EI until RSPEED_CMD_CHANGE_DONE.
27	**LcWaitForNonEiOnRxL0sExit.** Read-write. Defers Rx.L0s to Recovery transition until the PHY has seen a break from EI on its receivers. 0=Exit Rx.L0s if any condition to transition to Recovery is met. 1=Wait for PHY to detect an exit from EI on the receivers before begining transition out of Rx.L0s.
26	**LcWaitForLIdleInRIdle.** Read-write. Logical Idle counter control in Recovery.Idle. 0=Don't wait for required number of Logical Idles before sending Logical Idles. 1=Wait for required number of Logical Idles before sending Logical Idles.
25	**LcResetTsxCntInRconfigEn.** Read-write. TS Ordered Set Counter Control in Recovery.RcvrCfg. 0=No change in TSx COUNT if there is a change in the received TS Type in Recovery.RcvrCfg. 1=Reset TSx COUNT when there is a change in the received TS type in Recovery.RcvrCfg.

24	**LcMultReverseAttemptEn.** Read-write. Allow multiple reversal attempts when searching for widest possible link. 0=Allow only 1 link reversal in attempt to establish widest possible Link. 1=Allow 1 link reversal attempt after every LinkDown.
23	Reserved.
22:21	**LcDynLanesPwrState: unused link power state**. Read-write. Controls the state of unused links after a reconfiguration. Bits Definition Bits Definition 00b On 10b SB2 01b SB1 11b Off
20	**LcUpconfigCapable: upconfigure capable**. Read-only; Updated-by-hardware. 1=Both ends of the link are upconfigure capable. 0=Both ends of the link are not upconfigure capable.
19:14	Reserved.
13	**LcUpconfigureDis: upconfigure disable**. Read-write. 1=Disable link upconfigure.
12	**LcUpconfigureSupport: upconfigure support**. Read-write.
11	**LcShortReconfigEn: short re-configuration enable**. Read-write. 1=Enable short link re-configuration
10	**LcRenegotiateEn: link reconfiguration enable**. Read-write. 1=Enable link re-negotiation.
9	**LcRenegotiationSupport: re-negotiation support**. Read-only; Updated-by-hardware. 1=Link re-negotiation not supported by the downstream device.
8	**LcReconfigNow: re-configure link**. Read-write; Cleared-when-done. 1=Initiate link width change.
7	**LcReconfigArcMissingEscape**. Read-write. 1=Expedite transition from Recovery.Idle to Detect during a long reconfiguration.
6:4	**LcLinkWidthRd: current link width**. Read-only; Updated-by-hardware. Bits Definition Bits Definition 000b 0 100b 8 001b 1 101b 12 010b 2 110b 16 011b 4 111b Reserved
3	Reserved.
2:0	**LcLinkWidth: link width required**. Read-write. See: LcLinkWidthRd.

D2F[5:1]xE4_xA3 LC Number of FTS Control

Reset: 00FF_020Ch.

Bits	Description
31:10	Reserved.
9	**LcXmitFtsBeforeRecovery: transmit FTS before recovery**. Read-write. 1=Transmit FTS before recovery.
8:0	Reserved.

D2F[5:1]xE4_xA4 LC Link Speed Control

Reset: 0440_0100h.

Bits	Description
31:28	Reserved.
27	**LcMultUpstreamAutoSpdChngEn: enable multiple automatic speed changes**. Read-write. 1=Enable multiple automatic speed changes when D2F[5:1]xE4_xC0[StrapAutoRcSpeedNegotiationDis] == 0 and no failures occured in previous speed change attempts.
26:20	Reserved.
19	**LcOtherSideSupportsGen2: downstream link supports gen2**. Read-only; Updated-by-hardware. 1=The downstream link currently supports gen2.
18:15	Reserved.
14:13	Reserved.
12	**LcSpeedChangeAttemptFailed: speed change attempt failed**. Read-only; Updated-by-hardware. 1=LcSpeedChangeAttemptsAllowed has been reached.
11:10	Reserved.
9	**LcInitiateLinkSpeedChange: initiate link speed change**. Read-write; Cleared-when-done. 1=Initiate link speed negotiation.
8:7	Reserved.
6	**LcForceDisSwSpeedChange: force disable software speed changes**. Read-write. 1=Force the PCIe® core to disable speed changes initiated by private registers.
5:2	Reserved.
1	**LcGen3EnStrap: Gen3 PCIe® support enable**. Read-write. 1=Gen3 PCIe support enabled. 0=Gen3 PCIe support disabled.
0	**LcGen2EnStrap: Gen2 PCIe® support enable**. Read-write. 1=Gen2 PCIe support enabled. 0=Gen2 PCIe support disabled.

D2F[5:1]xE4_xA5 LC State 0

Cold reset: 0000_0000h.

Bits	Description
31:30	Reserved.
29:24	**LcPrevState3: previous link state 3**. Read-only; Updated-by-hardware. See: Table 81 [Link controller state encodings].
23:22	Reserved.
21:16	**LcPrevState2: previous link state 2**. Read-only; Updated-by-hardware. See: Table 81 [Link controller state encodings].
15:14	Reserved.
13:8	**LcPrevState1: previous link state 1**. Read-only; Updated-by-hardware. See: Table 81 [Link controller state encodings].

7:6	Reserved.
5:0	**LcCurrentState: current link state**. Read-only; Updated-by-hardware. See: Table 81 [Link controller state encodings].

Table 81: Link controller state encodings

Bits	Description	Bits	Description	Bits	Description
00h	s_Detect_Quiet.	12h	Rcv_L0_and_Tx_L0s.	24h	s_Rcvd_Loopback.
01h	s_Start_common_Mode.	13h	Rcv_L0_and_Tx_L0s_FTS.	25h	s_Rcvd_Loopback_Idle.
02h	s_Check_Common_Mode.	14h	Rcv_L0s_and_Tx_L0.	26h	s_Rcvd_Reset_Idle.
03h	s_Rcvr_Detect.	15h	Rcv_L0s_and_Tx_L0_Idle.	27h	s_Rcvd_Disable_Entry.
04h	s_No_Rcvr_Loop.	16h	Rcv_L0s_and_Tx_L0s.	28h	s_Rcvd_Disable_Idle.
05h	s_Poll_Quiet.	17h	Rcv_L0s_and_Tx_L0s_FTS.	29h	s_Rcvd_Disable.
06h	s_Poll_Active.	18h	s_L1_Entry.	2Ah	s_Detect_Idle.
07h	s_Poll_Compliance.	19h	s_L1_Idle.	2Bh	s_L23_Wait.
08h	s_Poll_Config.	1Ah	s_L1_Wait.	2Ch	Rcv_L0s_Skp_and_Tx_L0.
09h	s_Config_Step1.	1Bh	s_L1.	2Dh	Rcv_L0s_Skp_and_Tx_L0_Idle.
0Ah	s_Config_Step3.	1Ch	s_L23_Stall.	2Eh	Rcv_L0s_Skp_and_Tx_L0s.
0Bh	s_Config_Step5.	1Dh	s_L23_Entry.	2Fh	Rcv_L0s_Skp_and_Tx_L0_FTS.
0Ch	s_Config_Step2.	1Eh	s_L23_Entry.	30h	s_Config_Step2b.
0Dh	s_Config_Step4.	1Fh	s_L23_Ready.	31h	s_Recovery_Speed.
0Eh	s_Config_Step6.	20h	s_Recovery_lock.	32h	s_Poll_Compliance_Idle.
0Fh	s_Config_Idle.	21h	s_Recovery_Config.	33h	s_Rcvd_Loopback_Speed.
10h	Rcv_L0_and_Tx_L0.	22h	s_Recovery_Idle.	3Fh-34h	Reserved.
11h	Rcv_L0_and_Tx_L0_Idle.	23h	s_Training_Bit.		

D2F[5:1]xE4_xB1 LC Control 2

Reset: 9618_0280h.

Bits	Description
31:21	Reserved.
20	**LcBlockElIdleinL0: block electrical idle in l0**. Read-write. BIOS: 1. 1=Prevent electrical idle from causing the receiver to transition from L0 to L0s.
19	**LcDeassertRxEnInL0s: deassert RX_EN in L0s**. Read-write. 1=Turn off transmitters in L0s.
18:14	Reserved.
13:0	Reserved.

D2F[5:1]xE4_xB5 LC Control 3

Reset: 2850_5020h.

Bits	Description
31	Reserved.

30	**LcGoToRecovery: go to recovery**. Read-write. 1=Force link in the L0 state to transition to the Recovery state.
29:4	Reserved.
3	**LcRcvdDeemphasis: received deemphasis**. Read-only; Updated-by-hardware. Deemphasis advertised by the downstream device. 1=3.5dB. 0=6dB.
2:1	**LcSelectDeemphasisCntl: deemphasis control**. Read-write. Specifies the deemphasis used by the transmitter. Bits Definition 00b Use deemphasis from LcSelectDeemphasis. 01b Use deemphasis advertised by the downstream device. 10b 6dB. 11b 3.5dB.
0	**LcSelectDeemphasis: downstream deemphasis**. Read-write. Specifies the downstream deemphasis. 1=3.5dB. 0=6dB.

D2F[5:1]xE4_xC0 LC Strap Override

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15	**StrapAutoRcSpeedNegotiationDis: autonomous speed negotiation disable strap override**. Read-write. 1=Disable autonomous root complex speed negotiation to Gen2.
14	Reserved.
13	**StrapForceCompliance: force compliance strap override**. Read-write.
12:0	Reserved.

D2F[5:1]xE4_xC1 Root Port Miscellaneous Strap Override

Reset: 0000_0000h.

Bits	Description
31:6	Reserved.
5	**StrapLtrSupported**. Read-write.
4:3	**StrapObffSupported**. Read-write.
2	**StrapExtendedFmtSupported: Extended Fmt Supported strap override**. Read-write.
1	**StrapE2EPrefixEn: E2E Prefix En strap override**. Read-write.
0	**StrapReverseLanes: reverse lanes strap override**. Read-write.

D2F[5:1]xE4_xD0 Root Port ECC Skip OS Feature

Reset: 0000_0100h.

Bits	Description
31:16	**BchEccErrorStatus**. Read-write. Indicates that lane errors are above the specified threshold. (One bit per lane.)

15:8	**BchEccErrorThreshold**. Read-write. Error threshold.
7:1	Reserved.
0	**StrapBchEccEn**. Read-write.

D2F[5:1]xE4_xD2 PCIEP_HPGI_PRIVATE

Reset: 0000_0000h.

Bits	Description
31:7	Reserved.
6	**PresenceDetectStatePrivate**. Read-only. Unmasked version of PRESENCE_DETECT_STATE.
5:4	Reserved.
3	**PresenceDetectChangedPrivate**. Read-only. Unmasked version of PRESENCE_DETECT_CHANGED.
2:0	Reserved.

D2F[5:1]xE4_xDA PCIEP_HPGI

Reset: 0000_0000h.

Bits	Description
31:17	Reserved.
16	**RegHpgiPresenceDetectStateChangeEn**. Read-write. Asserted when both SMI and SCI work-around completed.
15	**HpgiRegPresenceDetectStateChangeStatus**. Read-only. Raw, unqualified, presence-detect-changed status from BIFcore, exported to wrapper.
14:12	Reserved.
11	**HpgiRegDeassertToSciStatus**. Read; Write-1-to-clear. Status bit indicating interrupt has been handled by OS.
10	**HpgiRegDeassertToSmiStatus**. Read; Write-1-to-clear. Status bit indicating interrupt has been handled by OS.
9	**HpgiRegAssertToSciStatus**. Read; Write-1-to-clear. Status bit indicating interrupt has been handled by SCI.
8	**HpgiRegAssertToSmiStatus**. Read; Write-1-to-clear. Status bit indicating interrupt has been handled by SMI.
7	**RegHpgiHook**. Read-write. Hot-plug work-around interrupt feature enable.
6:4	Reserved.
3	**RegHpgiDeassertToSciEn**. Read-write. Interrupt enable to qualify *_DEASSERT_TO_SCI_STATUS.
2	**RegHpgiDeassertToSmiEn**. Read-write. Interrupt enable to qualify *_DEASSERT_TO_SMI_STATUS.
1	**RegHpgiAssertToSciEn**. Read-write. Interrupt enable to qualify *_ASSERT_TO_SCI_STATUS.
0	**RegHpgiAssertToSmiEn**. Read-write. Interrupt enable to qualify *_ASSERT_TO_SMI_STATUS.

D2F[5:1]x100 Vendor Specific Enhanced Capability Register

Bits	Description
31:20	**NextPtr: next pointer**. Read-only. IF (D0F0xE4_x0140_00B0[StrapF0AerEn] == 1) THEN Reset: 150h. ELSE Reset: 000h. ENDIF.
19:16	**CapVer: capability version**. Read-only. Reset: 1h.
15:0	**CapID: capability ID**. Read-only. Reset: Bh.

D2F[5:1]x104 Vendor Specific Header Register

Reset: 0101_0001h.

Bits	Description
31:20	**VsecLen: vendor specific enhanced capability structure length**. Read-only. Defined the number of bytes of the entire vendor specific enhanced capability structure including the header.
19:16	**VsecRev: vendor specific enhanced capability version**. Read-only.
15:0	**VsecID: vendor specific enhanced capability ID**. Read-only.

D2F[5:1]x108 Vendor Specific 1 Register

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write. This field does not control any hardware.

D2F[5:1]x10C Vendor Specific 2 Register

Reset: 0000_0000h.

Bits	Description
31:0	**Scratch: scratch**. Read-write. This field does not control any hardware.

D2F[5:1]x128 Virtual Channel 0 Resource Status Register

Reset: 0002_0000h.

Bits	Description
31:18	Reserved.
17	**VcNegotiationPending: virtual channel negotiation pending**. Read-only; Updated-by-hardware. 1=Virtual channel negotiation in progress. This bit must be 0 before the virtual channel can be used.
16	**PortArbTableStatus: port arbitration table status**. Read-only.
15:0	Reserved.

D2F[5:1]x136 Virtual Channel 1 Resource Status Register

Reset: 0000_0002h.

Bits	Description
31:2	Reserved.
1	**VcNegotiationPending**. Read-only; Updated-by-hardware. This bit indicates whether the Virtual Channel negotiation (initialization or disabling) is in pending state. This bit must be 0 before the virtual channel can be used.
0	**PortArbTableStatus**. Read-only. This bit indicates the coherency status of the Port Arbitration Table associated with the VC resource.

D2F[5:1]x150 Advanced Error Reporting Capability

Bits	Description
31:20	**NextPtr: next pointer**. Read-only. IF (D0F0xE4_x0140_00B0[StrapF0AcsEn] == 1) THEN 2A0h. ELSE Reset: 000h. ENDIF.
19:16	**CapVer: capability version**. Read-only. Reset: 2h.
15:0	**CapID: capability ID**. Read-only. Reset: 1h.

D2F[5:1]x154 Uncorrectable Error Status

Cold reset: 0000_0000h.

Bits	Description
31:26	Reserved.
25	**TlpPrefixStatus: TLP prefix blocked status**. Read; Write-1-to-clear.
24	**AtomicOpEgressBlockedTLPStatus: atomic op egress blocked TLP status**. Read; Write-1-to-clear.
23	**McBlockedTLPStatus: MC blocked TLP status**. Read; Write-1-to-clear.
22	**UncorrInteralErrStatus: uncorrectable internal error status**. Read; Write-1-to-clear.
21	**AcsViolationStatus: access control service status**. Read; Write-1-to-clear.
20	**UnsuppReqErrStatus: unsupported request error status**. Read; Write-1-to-clear. The header of the unsupported request is logged.
19	**EcrcErrStatus: end-to-end CRC error status**. Read; Write-1-to-clear.
18	**MalTlpStatus: malformed TLP status**. Read; Write-1-to-clear. The header of the malformed TLP is logged.
17	**RcvOvflStatus: receiver overflow status**. Read-only.
16	**UnexpCplStatus: unexpected completion timeout status**. Read; Write-1-to-clear. The header of the unexpected completion is logged.
15	**CplAbortErrStatus: completer abort error status**. Read; Write-1-to-clear.
14	**CplTimeoutStatus: completion timeout status**. Read; Write-1-to-clear.

13	**FcErrStatus: flow control error status**. Read-only.
12	**PsnErrStatus: poisoned TLP status**. Read; Write-1-to-clear. The header of the poisoned transaction layer packet is logged.
11:6	Reserved.
5	**SurprdnErrStatus: surprise down error status**. Read-only. 0=Detection and reporting of surprise down errors is not supported.
4	**DlpErrStatus: data link protocol error status**. Read; Write-1-to-clear.
3:0	Reserved.

D2F[5:1]x158 Uncorrectable Error Mask

Cold reset: 0000_0000h.

Bits	Description
31:26	Reserved.
25	**TlpPrefixMask: TLP prefix blocked mask**. Read-only.
24	**AtomicOpEgressBlockedTLPMask: atomic op egress blocked TLP mask**. Read-only.
23	**McBlockedTLPMask: MC blocked TLP mask**. Read-only.
22	**UncorrInteralErrMask: uncorrectable internal error mask**. Read-write.
21	**AcsViolationMask: access control service mask**. Read-only. 1=ACS violation errors are not reported.
20	**UnsuppReqErrMask: unsupported request error mask**. Read-write. 1=Unsupported request errors are not reported.
19	**EcrcErrMask: end-to-end CRC error mask**. Read-write.
18	**MalTlpMask: malformed TLP mask**. Read-write. 1=Malformed TLP errors are not reported.
17	**RcvOvflMask: receiver overflow mask**. Read-only.
16	**UnexpCplMask: unexpected completion timeout mask**. Read-write. 1=Unexpected completion errors are not reported.
15	**CplAbortErrMask: completer abort error mask**. Read-write.
14	**CplTimeoutMask: completion timeout mask**. Read-write. 1=Completion timeout errors are not reported.
13	**FcErrMask: flow control error mask**. Read-only.
12	**PsnErrMask: poisoned TLP mask**. Read-write. 1=Poisoned TLP errors are not reported.
11:6	Reserved.
5	**SurprdnErrMask: surprise down error mask**. Read-only.
4	**DlpErrMask: data link protocol error mask**. Read-write. 1=Data link protocol errors are not reported.
3:0	Reserved.

D2F[5:1]x15C Uncorrectable Error Severity

Cold reset: 0006_2030h.

Bits	Description
31:26	Reserved.
25	**TlpPrefixSeverity: TLP prefix blocked severity**. Read-only.
24	**AtomicOpEgressBlockedTLPSeverity: atomic op egress blocked TLP severity**. Read-only.
23	**McBlockedTLPSeverity: MC blocked TLP severity**. Read-only.
22	**UncorrInteralErrSeverity: uncorrectable internal error severity**. Read-only.
21	**AcsViolationSeverity: access control service severity**. Read-only. 1=Fatal error. 0=Non-fatal error.
20	**UnsuppReqErrSeverity: unsupported request error severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
19	**EcrcErrSeverity: end-to-end CRC error severity**. Read-only.
18	**MalTlpSeverity: malformed TLP severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
17	**RcvOvflSeverity: receiver overflow severity**. Read-only.
16	**UnexpCplSeverity: unexpected completion timeout severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
15	**CplAbortErrSeverity: completer abort error severity**. Read-only.
14	**CplTimeoutSeverity: completion timeout severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
13	**FcErrSeverity: flow control error severity**. Read-only.
12	**PsnErrSeverity: poisoned TLP severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
11:6	Reserved.
5	**SurprdnErrSeverity: surprise down error severity**. Read-only.
4	**DlpErrSeverity: data link protocol error severity**. Read-write. 1=Fatal error. 0=Non-fatal error.
3:0	Reserved.

D2F[5:1]x160 Correctable Error Status

Cold reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15	**HdrLogOvflStatus: header log overflow status**. Read-only.
14	**CorrIntErrStatus: corrected internal error status**. Read; Write-1-to-clear.
13	**AdvisoryNonfatalErrStatus: advisory non-fatal error status**. Read; Write-1-to-clear. 1=A non-fatal unsupported request errors or a non-fatal unexpected completion errors occurred.
12	**ReplayTimerTimeoutStatus: replay timer timeout status**. Read; Write-1-to-clear.
11:9	Reserved.
8	**ReplayNumRolloverStatus: replay**. Read; Write-1-to-clear. 1=The same transaction layer packet has been replayed three times and has caused the link to re-train.

7	**BadDllpStatus: bad data link layer packet status**. Read; Write-1-to-clear. 1=A-link CRC error was detected.
6	**BadTlpStatus: bad transaction layer packet status**. Read; Write-1-to-clear. 1=A bad non-duplicated sequence ID or a link CRC error was detected.
5:1	Reserved.
0	**RcvErrStatus: receiver error status**. Read-only. 1=An 8b10b or disparity error was detected.

D2F[5:1]x164 Correctable Error Mask

Cold reset: 0000_6000h.

Bits	Description
31:16	Reserved.
15	**HdrLogOvflMask: header log overflow mask**. Read-only.
14	**CorrIntErrMask: corrected internal error mask**. Read-write.
13	**AdvisoryNonfatalErrMask: advisory non-fatal error mask**. Read-write. 1=Error is not reported.
12	**ReplayTimerTimeoutMask: replay timer timeout mask**. Read-write. 1=Error is not reported.
11:9	Reserved.
8	**ReplayNumRolloverMask: replay**. Read-write.1=Error is not reported.
7	**BadDllpMask: bad data link layer packet mask**. Read-write. 1=Error is not reported.
6	**BadTlpMask: bad transaction layer packet mask**. Read-write. 1=Error is not reported.
5:1	Reserved.
0	**RcvErrMask: receiver error mask**. Read-only. 1=Error is not reported.

D2F[5:1]x168 Advanced Error Control

Cold reset: 0000_0000h.

Bits	Description
31:12	Reserved.
11	**TlpPrefixLogPresent**. Read-only. IF (D2F[5:1]x7C[EndEndTlpPrefixSupported] == 0) THEN Reserved. ENDIF. 1=If FirstErrPtr is valid then the TLP Prefix Log register contains valid information.
10	**MultiHdrRecdEn**. Read-only. 1=Enables recording more than one error header.
9	**MultiHdrRecdCap**. Read-only. 1=Specifies that the function is capable of recording more than one error header.
8	**EcrcCheckEn: data link protocol error severity**. Read-write. 0=Specifies that End-to-end CRC generation is not supported.
7	**EcrcCheckCap: data link protocol error severity**. Read-only. 0=Specifies that end-to-end CRC check is not supported.
6	**EcrcGenEn: end-to-end CRC enable**. Read-only. 0=Specifies that End-to-end CRC generation is not supported.

5	**EcrcGenCap: end-to-end CRC capability**. Read-only. 0=Specifies that end-to-end CRC generation is not supported.
4:0	**FirstErrPtr: first error pointer**. Read-only. The First Error Pointer identifies the bit position of the first error reported in the Uncorrectable Error Status register.

D2F[5:1]x16C Header Log DW0

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 0 of the header.

D2F[5:1]x170 Header Log DW1

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 4 of the header.

D2F[5:1]x174 Header Log DW2

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 8 of the header.

D2F[5:1]x178 Header Log DW3

Cold reset: 0000_0000h.

Bits	Description
31:0	**TlpHdr: transaction layer packet header log**. Read-only. Contains the header for a transaction layer packet corresponding to a detected error. The upper byte represents byte 12 of the header.

D2F[5:1]x17C Root Error Command

Reset: 0000_0000h.

Bits	Description
31:3	Reserved.
2	**FatalErrRepEn: fatal error reporting enable**. Read-write. 1=Enables the generation of an interrupt when a fatal error is reported by any of the devices in the hierarchy associated with this Root Port.

1	**NonfatalErrRepEn: non-fatal error reporting enable**. Read-write. 1=Enables generation of an interrupt when a non-fatal error is reported by any of the devices in the hierarchy associated with this Root Port.
0	**CorrErrRepEn: correctable error reporting enable**. Read-write. 1=Enables generation of an interrupt when a correctable error is reported by any of the devices in the hierarchy associated with this Root Port.

D2F[5:1]x180 Root Error Status

Cold reset: 0000_0000h.

Bits	Description
31:27	**AdvErrIntMsgNum: advanced error interrupt message number**. Read-only.
26:7	Reserved.
6	**NFatalErrMsgRcvd: fatal error message received**. Read; Write-1-to-clear. Set to 1 when one or more fatal uncorrectable error messages have been received.
5	**NonFatalErrMsgRcvd: non-fatal error message received**. Read; Write-1-to-clear. Set to 1 when one or more non-fatal uncorrectable error messages have been received.
4	**FirstUncorrFatalRcvd: first uncorrectable fatal error message received**. Read; Write-1-to-clear. Set to 1 when the first uncorrectable error message received is for a fatal error.
3	**MultErrFatalNonfatalRcvd: ERR_FATAL/NONFATAL message received**. Read; Write-1-to-clear. Set when either a fatal or a non-fatal error is received and ErrFatalNonfatalRcvd is already set.
2	**ErrFatalNonfatalRcvd: ERR_FATAL/NONFATAL message received**. Read; Write-1-to-clear. Set when either a fatal or a non-fatal error is received and this bit is not already set.
1	**MultErrCorrRcvd: multiple ERR_COR messages received**. Read; Write-1-to-clear. Set when a correctable error message is received and ErrCorrRcvd is already set.
0	**ErrCorrRcvd: ERR_COR message received**. Read; Write-1-to-clear. Set when a correctable error message is received and this bit is not already set.

D2F[5:1]x184 Error Source ID

Cold reset: 0000_0000h.

Bits	Description
31:16	**ErrFatalNonfatalSrcID: ERR_FATAL/ERR_NONFATAL source identification**. Read-only. Loaded with the requestor ID indicated in the received ERR_FATAL or ERR_NONFATAL message when D2F[5:1]x180[ErrFatalNonfatalRcvd] is not already set.
15:0	**ErrCorlSrcID: ERR_COR source identification**. Read-only. Loaded with the requestor ID indicated in the received ERR_COR message when D2F[5:1]x180[ErrCorrRcvd] is not already set.

3.9 HD Audio Controller

3.9.1 Device 9h Function 2 (Audio Controller) Configuration Registers

D9F2x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 157Ah.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D9F2x04 Status/Command

Bits	Description
31	**DetectedParityErr**. Read; Write-1-to-clear. Reset: 0. This bit is set when a poisoned request or poisoned completion is received.
30	**SignaledSystemErr**. Read-only. Reset: 0. System Error is not supported.
29	**ReceivedMasterAbort**. Read; Write-1-to-clear. Reset: 0. This bit is set when a completion with UR status is received.
28	**ReceivedTargetAbort**. Read; Write-1-to-clear. Reset: 0. This bit is set when a completion with CA status is received.
27	**SignaledTargetAbort**. Read-only. Reset: 0.
26:25	Reserved.
24	**MasterDataParityErr**. Read; Write-1-to-clear. Reset: 0. This bit is set if ParityErrResp is set and either a poisoned completion is received, or a poisoned request is transmitted.
23:21	Reserved.
20	**CapList**. Read-only. Reset: 1. 1=Capability list supported.
19	**IntStatus**. Read-only. Reset: 0. 1=INTx message pending.
18:11	Reserved.
10	**InterruptDis**. Read-write. Reset: 0. 1=INTx interrupt message generation disabled.
9	Reserved.
8	**SerrEn**. Read-only. Reset: 0. 1=Enables reporting of non-fatal and fatal errors detected.
7	Reserved.
6	**ParityErrorResp**. Read-write. Reset: 0. 1=Allow logging of poisoned PCIe® Transaction Layer Packets (TLP) in MasterDataParityErr bit.
5:3	Reserved.
2	**BusMasterEn**. Read-only. Reset: 1. 1=Allows device to behave as a bus master.
1	**MemSpaceEn**. Read-write. Reset: 0. 1=Allows device to respond to memory space accesses.
0	Reserved.

D9F2x08 Class Code/Revision ID

Bits	Description
31:8	**ClassCode: class code**. Value: 04_0300h. Provides the audio controller class code.
7:0	**RevID: revision ID**. Value: 00h.

D9F2x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:24	Reserved.
23	**Multifunction**. Read-only. Indicates whether this is a multifunction device or not.
22:16	**HeaderType**. Read-only. Indicates Type 0 configuration space header format is supported.
15:0	Reserved.

D9F2x10 BAR0 Address Map

Bits	Description
31:14	**BaseAddr**. Read-write. Reset: 0. Indicates the Addr[31:14] of the base address. BAR 0 is 16KB is size.
13:4	Reserved.
3	**Prefetchable**. Read-only. Reset: 0. Indicates BAR 0 is prefetchable.
2:1	**Type: base address register type**. Value: 00b. 00b=32-bit base address register.
0	**Space**. Read-only. Value: 0. Indicates BAR 0 is a memory space.

D9F2x14 BAR1 Address Map

Reset: 0000_0000h.

Bits	Description
31:8	**BaseAddr**. Read-write. Indicates the Addr[31:8] of the base address. BAR 1 is 256B is size.
7:4	Reserved.
3	**Prefetchable**. Read-only. Indicates BAR 1 is non-prefetchable.
2:1	**Type**. Read-only. Indicates BAR 1 is in a 32-bit access space.
0	**Space**. Read-only. Indicates BAR 1 is a memory space.

D9F2x24 MSI-X BAR Address Map

Reset: 0000_0000h.

Bits	Description
31:13	**BaseAddr**. Read-write. Indicates the Addr[31:13] of the base address. MSI-X BAR is 8KB is size.

12:4	Reserved.
3	**Prefetchable**. Read-only. Indicates MSI-X BAR is non-prefetchable.
2:1	**Type**. Read-only. Indicates MSI-X BAR is in a 32-bit access space.
0	**Space**. Read-only. Indicates MSI-X BAR is a memory space.

D9F2x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr**. Read-only. Reset: 50h. Point to the offset of the function's MSI-X Capability list.

D9F2x3C Interrupt Line

Bits	Description
31:16	Reserved.
15:8	**InterruptPin**. Read-only. Reset: 1h. This field indicates the INTx line used to generate legacy interrupts. Bits — Description 00h — Device does not use an interrupt pin. 01h — INTA. 02h — INTB. 03h — INTC. 04h — INTD. FFh-05h — Reserved.
7:0	**InterruptLine**. Read-write. Reset: 0. Indicates which input of the system interrupt controllers is connected to device's interrupt pin.

D9F2x40 ACGAZ Mirror Reg Ctrl 0

Bits	Description
31:8	**ClassCodeW**. Read-write. Reset: 04_0300h. Program the ClassCode in offset 08h.
7:0	**RevidW**. Read-write. Reset: 00h. Program the RevID in offset 08h.

D9F2x44 ACGAZ Mirror Reg Ctrl 1

Bits	Description
31:16	**DeviceIdW**. Read-write. Reset: 157Ah. Program the DeviceId in offset 00h.
15	**SetPmeStatus**. Write-only. Reset: 0. Program the register to set the PME status register in the PCI capabilities register.
14:10	**PmeSupportW**. Read-write. Reset: 19h. Program the PME Support field in the PCI PM capabilities register.

9	**PciFlrSupW**. Read-write. Reset: 0. Program the FLR Support field in the PCI advanced features capabilities register.
8	**PcieFlrSupW**. Read-write. Reset: 0. Program the FLR Support field in the PCIe® device capabilities register.
7:0	**PmNxtPtrW**. Read-write. Reset: 00h. Program the next item in the function's capability list to point to the PM Capability in offset 60h.

D9F2x50 ACGAZ MSI-X Capability Register

Bits	Description
31	**MsiXEn**. Read-write. Reset: 0. 1=Enable MSI-X functionality.
30	**FuncMask**. Read-write. Reset: 0. 1=All of the vectors associated with PSP are masked, regardless of their per-vector Mask bit states. 0=Each vector's Mask bit determines whether the vector is masked or not.
29:27	Reserved.
26:16	**TableSize**. Read-only. Reset: 0h. Indicates the MSI-X TableSize N, which is encoded as N-1.
15:8	**NxtPtr**. Read-only. Reset: 5Ch. Indicates the next item in the function's capability list.
7:0	**CapId**. Read-only. Reset: 11h. Indicates this linked-list item as MSI-X registers.

D9F2x54 ACGAZ MSI-X Table Structure Register

Bits	Description
31:3	**TableOffset**. Read-only. Reset: 0. Offset the TableBir as the base of the MSI-X table.
2:0	**TableBir**. Read-only. Reset: 5h. Indicates the MSI-X table uses the base address register specified: Bits — Description 000b — Offset 10h. 001b — Offset 14h. 010b — Offset 18h. 011b — Offset 1Ch. 100b — Offset 20h. 101b — Offset 24h. 111b-110b — Reserved.

D9F2x58 ACGAZ PBA Structure Register

Bits	Description
31:3	**PbaOffset**. Read-only. Reset: 200h. Offset the PbaBir as the base of the MSI-X PBA.
2:0	**PbaBir**. Read-only. Reset: 5h. Indicates the MSI-X pending bit array (PBA) uses the base address register specified: Bits Description 000b Offset 10h. 001b Offset 14h. 010b Offset 18h. 011b Offset 1Ch. 100b Offset 20h. 101b Offset 24h. 111b-110b Reserved.

D9F2x5C ACGAZ HyperTransport™ MSI Capability Register

Bits	Description
31:27	**CapType**. Read-only. Reset: 15h. Indicates this is an MSI mapping capability block.
26:18	Reserved.
17	**Fixd**. Read-only. Reset: 1h. Indicates the address for mapping MSIs is fixed at 0000_0000_FEEX_XXXXh.
16	**MapEn**. Read-only. Reset: 1h. Indicates MSI mapping is active.
15:8	**NxtPtr**. Read-only. Reset: 60h. The next item in the function's capability list: PM Capability.
7:0	**CapId**. Read-only. Reset: 8h. This linked list item is the HyperTransport™ capability registers.

D9F2x60 ACGAZ PM Capability Register

Bits	Description
31:27	**PmeSupport**. Read-only. Reset: 0h. PME generation is not supported in any state.
26	**D2Support**. Read-only. Reset: 0h. D2 is not supported.
25	**D1Support**. Read-only. Reset: 0h. D1 is not supported.
24:22	**AuxCurrent**. Read-only. Reset: 0h. PME generation is not supported in any state.
21	**Dsi**. Read-only. Reset: 0h. No device specific initialization required.
20	Reserved.
19	**PmeClock**. Read-only. Reset: 0h. PME generation is not supported in any state.
18:16	**Version**. Read-only. Reset: 3h. Indicates revision 1.2 of the PCI PM Interface Specification.

15:8	**NxtPtr**. Read-write. Reset: 0h. The next item in the function's capability list: Bits — Description 0h — NULL for final item in the list. 67h-1h — Reserved. 68h — PCIe® capability structure. A3h-69h — Reserved. A4h — PCI advanced features capability. FFh-A5h — Reserved.
7:0	**CapId**. Read-only. Reset: 1h. This linked list item is the PCIe power management registers.

D9F2x68 ACGAZ PCIe® Capability Register

Bits	Description
31:30	Reserved.
29:25	**IntMsgNum**. Read-only. Reset: 0. This field indicates which MSI-X vector is used for the interrupt message generated in association with any of the status bits of this capability structure.
24	Reserved.
23:20	**FuncType**. Read-only. Reset: 9h. Indicates this function is a root complex integrated endpoint.
19:16	**Version**. Read-only. Reset: 2h. Indicates PCI-SIG defined PCIe capability structure version number.
15:8	**NxtPtr**. Read-only. Reset: 0. The next item in the function's capability list: NULL - End of capability list.
7:0	**CapId**. Read-only. Reset: 10h. This linked list item is the PCIe capability registers.

D9F2x6C ACGAZ PCIe® Device Capability

Bits	Description
31:29	Reserved.
28	**FlrSupport**. Read-only. Reset: 0. Indicates whether function level reset is supported.
27:16	Reserved.
15	**RoleErrorRpt**. Read-only. Reset: 1. Indicates error reporting ECN is implemented.
14:12	Reserved.
11:9	**L1AcceptedLat**. Read-only. Reset: 7h. There is no limit on acceptable latency due to transition from L1 to L0 state.
8:6	**L0AcceptedLat**. Read-only. Reset: 7h. There is no limit on acceptable total latency due to transition from L0s to L0 state.
5	**ExtendTagField**. Read-only. Reset: 1. The maximum supported size of tag field is 8-bit.
4:3	**PhantomFunction**. Read-only. Reset: 0. Phantom function is not supported.
2:0	**MaxPayloadSize**. Read-only. Reset: 0. The maximum payload size supported is 128B.

D9F2x70 ACGAZ PCIe® Device Ctrl Status

Bits	Description
31:22	Reserved.
21	**TransactionPending**. Read-only. Reset: 0. Indicates the function has issued a non-posted request that has not been completed yet.
20	**AuxPowerDetected**. Read-only. Reset: 0. Aux power reporting is not supported.
19	**UrErrDetected**. Read; Write-1-to-clear. Reset: 0. Indicates an unsupported request has been detected.
18	**FatalErrDetected**. Read; Write-1-to-clear. Reset: 0. Indicates a fatal error has been detected.
17	**NonfatalErrDetected**. Read; Write-1-to-clear. Reset: 0. Indicates a non-fatal error has been detected.
16	**CorrErrDetected**. Read-only. Reset: 0. Function does not generate correctable error.
15	**InitFlr**. Write-only. Reset: 0. Initiates Function Level Reset (FLR) to the function when writing a 1 to this field. Reading from this field always return 0.
14:12	**MaxReadSize**. Read-only. Reset: 0. Hardwired to 128B max Read Request size.
11	**EnableNsnoop**. Read-write. Reset: 1. Allow function to set the No Snoop bit in the initiated transaction attribute field.
10	**AuxPowerPmEn**. Read-only. Reset: 0. Aux power capability is not supported.
9	**PhantomFuncEn**. Read-only. Reset: 0. Phantom function is not supported.
8	**ExtendTagField**. Read-write. Reset: 1. 0=5-bit tag field. 1=8-bit tag field.
7:5	**MaxPayloadSize**. Read-only. Reset: 0.
4	**EnableRo**. Read-write. Reset: 1. Allow function to set Relaxed Ordering bit in the initiated transaction attribute field.
3	**UrErrRptEn**. Read-only. Reset: 0.
2	**FatalErrRptEn**. Read-only. Reset: 0.
1	**NonfatalErrRptEn**. Read-only. Reset: 0.
0	**CorrErrRptEn**. Read-only. Reset: 0.

D9F2x8C ACGAZ PCIe® Device Capability 2

Bits	Description
31:22	Reserved.
21	**End2endTlpprefix**. Read-only. Reset: 0.
20	**ExtendFmt**. Read-only. Reset: 0.
19:18	**ObffSupport**. Read-only. Reset: 0.
17:14	Reserved.
13:12	**TphCmpl**. Read-only. Reset: 0.
11	**LtrMechanism**. Read-only. Reset: 0.
10	Reserved.
9	**Cas128bCmpl**. Read-only. Reset: 0.

8	**Atomic64bCmpl**. Read-only. Reset: 0.
7	**Atomic32bCmpl**. Read-only. Reset: 0.
6:4	Reserved.
3:0	**CmplTimeoutRange**. Read-only. Reset: 0. Function does not support completion timeout programming.

D9F2x90 ACGAZ PCIe® Device Ctrl Status 2

Bits	Description
31:15	Reserved.
14:13	**ObffEn**. Read-only. Reset: 0.
12:11	Reserved.
10	**LtrMechanismEn**. Read-only. Reset: 0.
9	**IdoCmplEn**. Read-only. Reset: 0.
8	**IdoReqEn**. Read-only. Reset: 0.
7	**AtomicEgressBlock**. Read-only. Reset: 0.
6	**AtomicReqEn**. Read-only. Reset: 0.
5:4	Reserved.
3:0	**CmplTimeoutValue**. Read-only. Reset: 0.

D9F2xA4 ACGAZ PCI Advance Features Capability Register

Bits	Description
31:26	Reserved.
25	**FlrCap**. Read-only. Reset: 0. Indicates whether function level reset is supported.
24	**TpCap**. Read-only. Reset: 1. Indicates transaction pending bit is supported.
23:16	**Length**. Read-only. Reset: 6h. Indicates the length of the PCI Advance Features structure.
15:8	**NxtPtr**. Read-only. Reset: 0. NULL - end of capability list.
7:0	**CapId**. Read-only. Reset: 13h. Linked list item is PCI Advance Features capability.

D9F2xA8 ACGAZ PCI Advance Features Ctrl Status

Bits	Description
31:9	Reserved.
8	**TransactionPending**. Read-only. Reset: 0. Indicates the function has issued a non-posted request that has not completed yet.
7:1	Reserved.
0	**InitFlr**. Write-only. Reset: 0. Indicates FLR to the function when writing a 1h to this field. Reading from this field always returns 0.

D9F2xE8 NB ACG CMN Config Index(NB ACG CMN INDEX)

The index/data pair registers, D9F2xE8 and D9F2xEC are used to access the registers at D9F2xEC_x[FF:00]. To access any of these registers, the address is first written into the index register, D9F2xE8, and then the data is read from or written to the data register D9F2xEC without an intervening read of D9F2xE8. A read of D9F2xE8 causes the prior index value written to be cleared following the read.

Bits	Description
31:8	Reserved.
7:0	**NbAcgCmnIndAddr**. Read-write. Reset: 0.

D9F2xEC NB ACG CMN Config Data(NB ACG CMN DATA)

Reset: 0000_0000h.

Bits	Description
31:0	**NbAcgCmnData**. Read-write.

D9F2xEC_x4A ACG LCLK Clock Gating Control0

Reset: 7F3F_8100h.

Bits	Description
31	Reserved.
30	**SoftOverrideClk0**. Read-write.
29	**SoftOverrideClk1**. Read-write.
28	**SoftOverrideClk2**. Read-write.
27	**SoftOverrideClk3**. Read-write.
26	**SoftOverrideClk4**. Read-write.
25	**SoftOverrideClk5**. Read-write.
24	**SoftOverrideClk6**. Read-write.
23:22	Reserved.
21	**RampDisClk0**. Read-only. Disable ramping feature for CLK0.
20	**RampDisClk1**. Read-only. Disable ramping feature for CLK1.
19	**RampDisClk2**. Read-only. Disable ramping feature for CLK2.
18	**RampDisClk3**. Read-only. Disable ramping feature for CLK3.
17	**RampDisClk4**. Read-only. Disable ramping feature for CLK4.
16	**RampDisClk5**. Read-only. Disable ramping feature for CLK5.
15	**RampDisClk6**. Read-only. Disable ramping feature for CLK6.

14:12	**DivId**. Read-only. Output clock during idle. Bits: Description 000b: Clock turn off. 001b: Divide by 2. 010b: Divide by 4. 011b: Divide by 8. 100b: Divide by 16. 101b: Divide by 32. 111b-110b: Reserved.
11:4	**OffHysteresis**. Read-write. Hysteresis for local clock gating to filter out small turn off changes.
3:0	**OnDelay**. Read-only. Delay for local clock gating turn on.

D9F2xEC_x4C ACG AXICLK Clock Gating Control0

Reset: 7F3F_8100h.

Bits	Description
31	Reserved.
30	**SoftOverrideClk0**. Read-write.
29	**SoftOverrideClk1**. Read-write.
28	**SoftOverrideClk2**. Read-write.
27	**SoftOverrideClk3**. Read-write.
26	**SoftOverrideClk4**. Read-write.
25	**SoftOverrideClk5**. Read-write.
24	**SoftOverrideClk6**. Read-write.
23:22	Reserved.
21	**RampDisClk0**. Read-only. Disable ramping feature for CLK0.
20	**RampDisClk1**. Read-only. Disable ramping feature for CLK1.
19	**RampDisClk2**. Read-only. Disable ramping feature for CLK2.
18	**RampDisClk3**. Read-only. Disable ramping feature for CLK3.
17	**RampDisClk4**. Read-only. Disable ramping feature for CLK4.
16	**RampDisClk5**. Read-only. Disable ramping feature for CLK5.
15	**RampDisClk6**. Read-only. Disable ramping feature for CLK6.
14:12	**DivId**. Read-only. Output clock during idle. Bits: Description 000b: Clock turn off. 001b: Divide by 2. 010b: Divide by 4. 011b: Divide by 8. 100b: Divide by 16. 101b: Divide by 32. 111b-110b: Reserved.
11:4	**OffHysteresis**. Read-write. Hysteresis for local clock gating to filter out small turn off changes.
3:0	**OnDelay**. Read-only. Delay for local clock gating turn on.

D9F2xF4_x0F ACGAZ Downstream BAR Mapping

Reset: 0000_0000h.

Bits	Description
31:2	Reserved.
1	**Bar1MapLock**. Read-write. 1=Bar1MapLock and Bar1MapLo/Hi are locked until the next system reset.
0	**Bar0MapLock**. Read-write. 1=Bar0MapLock and Bar0MapLo/Hi are locked until the next system reset.

D9F2xF4_x1[3:0] ACGAZ Downstream BAR

Reset: 0000_0000h.

Table 82: Register Mapping for D9F2xF4_x1[3:0]

Register	Function	Register	Function
D9F2xF4_x10	BAR 0 MapLo	D9F2xF4_x12	BAR 1 MapLo
D9F2xF4_x11	BAR 0 MapHi	D9F2xF4_x13	BAR 1 MapHi

Bits	Description
31:0	**BarMap**. Read-write. Specifies the address where PCIe® BAR n base address maps in AXI addressing space.

D9F2xF4_x20 ACGAZ Upstream Control

Reset: 0020_0000h.

Bits	Description
31:22	Reserved.
21:16	**RspbufSize**. Read-write. This is the number of DMA Read Response buffers available for tag/response allocation. (Each buffer is 64 B.)
15:13	Reserved.
12:11	**Aper1NsnoopMap**. Read-write. Upstream PCIe® NoSnoop attribute control for DMA targeting Aper1 window. Bits — Description 00b — Always set NoSnoop to 0b. 01b — Always set NoSnoop to 1b. 10b — Set NoSnoop=1b if both AxCache[3:2]=00b. 11b — Reserved.
10	Reserved.

9:8	**Aper0NsnoopMap**. Read-write. Upstream PCIe NoSnoop attribute control for DMA targeting Aper0 window. Bits: Description 00b: Always set NoSnoop to 0b. 01b: Always set NoSnoop to 1b. 10b: Set NoSnoop=1b if both AxCache[3:2]=00b. 11b: Reserved.
7:6	**Aper1RoMap**. Read-write. Upstream PCIe Relaxed Ordering attribute control for DMA targeting Aper1 window. Bits: Description 00b: Always set RO to 0b. 01b: Always set RO to 1b. 10b: Set RO = 0b if AxCache == 0000b or 0010b, otherwise RO = 1b. 11b: Reserved.
5:4	**Aper0RoMap**. Read-write. Upstream PCIe Relaxed Ordering attribute control for DMA targeting Aper0 window. Bits: Description 00b: Always set RO to 0b. 01b: Always set RO to 1b. 10b: Set RO = 0b if AxCache == 0000b or 0010b, otherwise RO = 1b. 11b: Reserved.
3	**Aper1En**. Read-write. 1=AXI DMA window Aper1 is enabled.
2	**Aper0En**. Read-write. 1=AXI DMA window Aper0 is enabled.
1	**Aper1Lock**. Read-write. 1=Aper1Lock, Aper1En, Aper1BaseLo/Hi and Aper1MapLo/Hi are locked until the next system reset.
0	**Aper0Lock**. Read-write. 1=Aper0Lock, Aper0En, Aper0BaseLo/Hi and Aper0MapLo/Hi are locked until the next system reset.

D9F2xF4_x21 ACGAZ Upstream PASID Control

Reset: 0000_0000h.

Bits	Description
31:25	Reserved.
24	**PasidEn**. Read-write. 1=Enable the use of PASID on DMA requests.
23:0	**Pasid**. Read-write. PASID.

D9F2xF4_x2[9:8] ACGAZ Upstream Aper0 BAR

Reset: 0000_0000h.

Table 83: Register Mapping for D9F2xF4_x2[9:8]

Register	Function
D9F2xF4_x28	Aper 0 BaseLo
D9F2xF4_x29	Aper 0 BaseHi

Table 84: Reserved field mappings for D9F2xF4_x2[9:8]

Register	Bits
	23:0
D9F2xF4_x28	Reserved
D9F2xF4_x29	-

Bits	Description
31:0	**Aper0Base**. Read-write. Base address of the AXI DMA window Aper0.

D9F2xF4_x2[B:A] ACGAZ Upstream Aper0 Limit Register

Reset: 0000_00FFh.

Table 85: Register Mapping for D9F2xF4_x2[B:A]

Register	Function
D9F2xF4_x2A	Aper 0 LimitLo
D9F2xF4_x2B	Aper 0 LimitHi

Table 86: Reserved field mappings for D9F2xF4_x2[B:A]

Register	Bits
	23:0
D9F2xF4_x2A	Reserved
D9F2xF4_x2B	-

Bits	Description
31:0	**Aper0Limit**. Read-write. Limit address of the AXI DMA window Aper0.

D9F2xF4_x2[D:C] ACGAZ Upstream Aper0 Map Register

Reset: 0000_0000h.

Table 87: Register Mapping for D9F2xF4_x2[D:C]

Register	Function
D9F2xF4_x2C	Aper 0 MapLo
D9F2xF4_x2D	Aper 0 MapHi

Table 88: Reserved field mappings for D9F2xF4_x2[D:C]

Register	Bits
	23:0
D9F2xF4_x2C	Reserved
D9F2xF4_x2D	-

Bits	Description
31:0	**Aper0Map**. Read-write. Specifies the PCIe base address where the AXI DMA window Aper0 base address maps.

D9F2xF4_x48 ACGAZ Interrupt Polarity Control Register

Reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**InterruptPolarity**. Read-write. BIOS: 1. 1=AXI interrupt line is active low. 0=AXI interrupt line is active high. The polarity of the interrupt must be programmed before D9F2xF4_x49[InterruptEnable] is enabled.

D9F2xF4_x49 ACGAZ Interrupt Enable Register

Reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**InterruptEnable**. Read-write. 1=AXI interrupt line is enabled. 0=AXI interrupt line is disabled. D9F2xF4_x48[InterruptPolarity] must be programmed before the interrupt is enabled.

3.10 Device 18h Function 0 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D18F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 15B0h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D18F0x04 Status/Command

Bits	Description
31:16	**Status**. Read-only. Value: 0010h. Bit[20] is set to indicate the existence of a PCI-defined capability block.
15:0	**Command**. Read-only. Value: 0000h.

D18F0x08 Class Code/Revision ID

Bits	Description
31:8	**ClassCode**. Read-only. Value: 06_0000h. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only. Value: 00h.

D18F0x0C Header Type

Read-only. Value: 0080_0000h.

Bits	Description
31:0	**HeaderTypeReg**. These bits are fixed at their default values. The header type field indicates that there are multiple functions present in this device.

D18F0x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Value: 00h.

D18F0x[5C:40] Routing Table

Reset: 0004_0201h.

Table 89: Register Mapping for D18F0x[5C:40]

Register	Function
D18F0x40	Node 0
D18F0x[5C:44]	Reserved

Bits	Description
31:0	Reserved.

D18F0x60 Node ID

Bits	Description
31:21	Reserved.
20:16	**CpuCnt[4:0]: CPU count bits[4:0]**. Read-write. Reset: 0. Specifies the number of cores to be enabled (the boot core plus those cores enabled through D18F0x1DC[CpuEn]). Bits — Description 00h — 1 core 02h-01h — <CpuCnt[4:0] + 1> cores 03h — 4 cores 1Fh-04h — Reserved
15:0	Reserved.

D18F0x64 Unit ID

Reset: 0000_00E0h.

Bits	Description
31:16	Reserved.
15:11	**PspUnitId: PSP Unit ID**. IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Specifies the Unid ID used by the PSP. Only upstream requests that match this Unit ID is allowed access to certain protected regions in DRAM.
10:8	Reserved.
7:6	**HbUnit: host bridge Unit ID**. Read-only. Specifies the coherent link Unit ID of the host bridge used by the coherent fabric.
5:4	**MctUnit: memory controller Unit ID**. Read-only. Specifies the coherent link Unit ID of the memory controller.
3:0	Reserved.

D18F0x68 Link Transaction Control

Bits	Description
31:24	Reserved.
23	**InstallStateS**. Read-write. Reset: 0. 1=Forces the default read block (RdBlk) install state to be shared instead of exclusive.
22:21	**DsNpReqLmt: downstream non-posted request limit**. Read-write. Reset: 00b. BIOS: 10b. This specifies the maximum number of downstream non-posted requests issued by core(s) which may be outstanding on the IO links attached to this node at one time. Bits Description 00b No limit 01b limited to 1 10b limited to 4 11b limited to 8
20	**SeqIdSrcNodeEn: sequence ID source node enable**. Read-write. Reset: 0. 1=The source node ID of requests is provided in the SeqID field of the corresponding downstream IO link request packets. This may be useful for debug applications, in order to match downstream packets with their originating node. For normal operation, this bit should be cleared. Correct ordering of requests between different nodes is not ensured when this bit is set. Semaphore sharing between differing nodes may not work properly in systems which are capable of processing IO requests with differing non-zero SeqIds out of request order.
19	**ApicExtSpur: APIC extended spurious vector enable**. Read-write. Reset: 0. This enables the extended APIC spurious vector functionality; it affects APICF0[Vector]. 0=The lower 4 bits of the spurious vector are read-only 1111b. 1=The lower 4 bits of the spurious vector are writable.
18	**ApicExtId: APIC extended ID enable**. Read-write. Reset: 0. Enables the extended APIC ID functionality. 0=APIC ID is 4 bits. 1=APIC ID is 8 bits.
17	**ApicExtBrdCst: APIC extended broadcast enable**. Read-write. Reset: 0. Enables the extended APIC broadcast functionality. 0=APIC broadcast is 0Fh. 1=APIC broadcast is FFh. If ApicExtBrdCst == 1 then software must assert ApicExtId.
16	**LintEn: local interrupt conversion enable**. Read-write. Reset: 0. 1=Enables the conversion of broadcast ExtInt and NMI interrupt requests to LINT0 and LINT1 local interrupts, respectively, before delivering to the local APIC. This conversion only takes place if the local APIC is hardware enabled. LINT0 and LINT1 are controlled by APIC3[60:50]. 0=ExtInt/NMI interrupts delivered unchanged.
15	**LimitCldtCfg: limit coherent link configuration space range**. Read-write. Reset: 0. BIOS: 1.
14:12	Reserved.
11	**RespPassPW: response PassPW**. Read-write. Reset: 0. BIOS: 1. 1=The PassPW bit in all downstream link responses is set, regardless of the originating request packet. This technically breaks the PCI ordering rules but it is not expected to be an issue in the downstream direction. Setting this bit improves the latency of upstream requests by allowing the downstream responses to pass posted writes. 0=The PassPW bit in downstream responses is based on the RespPassPW bit of the original request.
10	**DisFillP: disable fill probe**. Read-write. Reset: 0. Controls probes for core-generated fills. 0=Probes issued for cache fills. 1=Probes not issued for cache fills.

9	**DisRmtPMemC: disable remote probe memory cancel**. Read-write. Reset: 0. 1=Only probed caches on the same node as the target memory controller may generate MemCancel coherent link packets. MemCancels are used to attempt to save DRAM and/or link bandwidth associated with the transfer of stale DRAM data. 0=Probes hitting dirty blocks may generate MemCancel packets, regardless of the location of the probed cache.
8	**DisPMemC: disable probe memory cancel**. Read-write. Reset: 0. Controls generation of MemCancel coherent link packets. MemCancels are used to attempt to save DRAM and/or coherent link bandwidth associated with the transfer of stale DRAM data. 0=Probes hitting dirty blocks of the core cache may generate MemCancel packets. 1=Probes may not generate MemCancel packets.
7	**CPURdRspPassPW: CPU read response PassPW**. Read-write. Reset: 0. 1=Read responses to core-generated reads are allowed to pass posted writes. 0=Core responses do not pass posted writes. This bit is not expected to be set. This bit may only be set during the boot process.
6	**CPUReqPassPW: CPU request PassPW**. Read-write. Reset: 0. 1=Core-generated requests are allowed to pass posted writes. 0=Core requests do not pass posted writes. This bit is not expected to be set. This bit may only be set during the boot process.
5	Reserved.
4	**DisMTS: disable memory controller target start**. Read-write. Reset: 0. BIOS: 1. 1=Disables use of TgtStart. TgtStart is used to improve scheduling of back-to-back ordered transactions by indicating when the first transaction is received and ordered at the memory controller.
3	**DisWrDwP: disable write doubleword probes**. Read-write. Reset: 0. BIOS: 0. 1=Disables generation of probes for core-generated, WrSized doubleword commands.
2	**DisWrBP: disable write byte probes**. Read-write. Reset: 0. BIOS: 0. 1=Disables generation of probes for core-generated, WrSized byte commands.
1	**DisRdDwP: disable read doubleword probe**. Read-write. Reset: 0. BIOS: 0. 1=Disables generation of probes for coregenerated, RdSized doubleword commands.
0	**DisRdBP: disable read byte probe**. Read-write. Reset: 0. BIOS: 0. 1=Disables generation of probes for core-generated, RdSized byte commands.

D18F0x6C Link Initialization Control

Bits	Description
31	Reserved.
30	**RlsLnkFullTokCntImm**: **release upstream full token count immediately**. Read-write. Cold reset: 0. BIOS: 1 after buffer counts have been programmed. 1=Apply buffer counts programmed in D18F0x[F0,D0,B0,90] and D18F0x[F4,D4,B4,94] immediately without requiring warm reset. Once this bit is set, additional changes to the buffer counts only take effect upon warm reset.
29	Reserved.
28	**RlsIntFullTokCntImm: release internal full token count immediately**. Read-write. Cold reset: 0. BIOS: 1 after buffer counts have been programmed. 1=Apply buffer counts programmed in D18F3x6C, D18F3x70, D18F3x74, D18F3x78, D18F3x7C, D18F3x140, D18F3x144, D18F3x1[54,50,4C,48], D18F3x17C, and D18F3x1A0 immediately without requiring warm reset. Once this bit is set, additional changes to the buffer counts only take effect upon warm reset.

27	**ApplyIsocModeEnNow**. Read-write. Cold reset: 0. BIOS: 1 after RlsLnkFullTokCntImm and RlsIntFullTokCntImm have been set. 1=Apply the programmed value in D18F0x[E4,C4,A4,84][IsocEn] immediately without requiring warm reset. This bit may only be set if RlsLnkFullTokCntImm and RlsIntFullTokCntImm are set and isochronous buffers have been allocated. IF (ApplyIsocModeEnNow) THEN (D18F3x1[54,50,4C,48][IsocPreqTok0] > 0).
26:11	Reserved.
10:9	**BiosRstDet[2:1]: BIOS reset detect bits[2:1]**. See: BiosRstDet[0].
8:7	Reserved.
6	**InitDet: CPU initialization command detect**. Read-write. Reset: 0. This bit may be used by software to distinguish between an INIT and a warm or cold reset by setting it to a 1 before an initialization event is generated. This bit is cleared by RESET_L but not by an INIT command.
5	**BiosRstDet[0]: BIOS reset detect bit[0]**. Read-write. Cold reset: 0. BiosRstDet[2:0] = {BiosRstDet[2:1], BiosRstDet[0]}. May be used to distinguish between a reset event generated by the BIOS versus a reset event generated for any other reason by setting one or more of the bits to a 1 before initiating a BIOS-generated reset event.
4	**ColdRstDet: cold reset detect**. Read-write. Cold reset: 0. This bit may be used to distinguish between a cold versus a warm reset event by setting the bit to a 1 before an initialization event is generated.
3:1	Reserved.
0	**RouteTblDis: routing table disable**. Read-write. Reset: 1. BIOS: 0.

D18F0x[E4,C4,A4,84] Link Control

Table 90: Register Mapping for D18F0x[E4,C4,A4,84]

Register	Function
D18F0x84	ONION Link
D18F0x[E4,C4,A4]	Reserved

This register is derived from the link control register defined in the link specification.

Bits	Description
31:16	Reserved.
15	**Addr64BitEn: 64-bit address packet enable**. Read-write. Cold reset: 0. 1=Requests to addresses greater than FF_FFFF_FFFFh are supported by this IO link. 0=Requests to addresses greater than FF_FFFF_FFFFh are master aborted as if the end of chain was reached. BIOS is required to ensure that the link-specification-defined “64-bit Address Feature” bit in the device on the other side of the link is set prior to setting this bit. For coherent links, this bit is unused. D18F0x68[CHtExtAddrEn] is required to be set if this bit is set for any IO link. The link specification indicates that this bit is cleared by a warm reset; therefore this bit may be in a different state than an IO device on the other side of the link after a warm reset; care should be taken by BIOS to place devices on both sides of the link in the same state after a warm reset, before any packets to the high-order addresses enabled by this bit are generated.
14:13	Reserved.

12	**IsocEn: isochronous flow-control mode enable**. Read-write. Cold reset: 0. BIOS: 1 if the link is an ONION Link. This bit is set to place the link into isochronous flow-control mode (IFCM), as defined by the link specification. 1=IFCM. 0=Normal flow-control mode. See D18F0x6C[ApplyIsocModeEnNow].
11:5	Reserved.
4	**LinkFail: link failure**. Read; Set-by-hardware; Write-1-to-clear. Cold reset: 0. This bit is set high by the hardware if a Sync flood is received by the link. See 2.16.1.8.1 [Common Diagnosis Information].
3:0	Reserved.

D18F0x[EC,CC,AC,8C] Link Feature Capability

This register is derived from the link feature capability register defined in the link specification. Unless otherwise specified: 0=The feature is not supported; 1=The feature is supported.

Table 91: Register Mapping for D18F0x[EC,CC,AC,8C]

Register	Function
D18F0x8C	ONION Link
D18F0x[EC,CC,AC]	Reserved

Bits	Description
31:6	Reserved.
5	**UnitIdReOrderDis: UnitID reorder disable**. Read-write. Reset: 0. 1=Upstream reordering for different UnitIDs is not supported (i.e., all upstream packets are ordered as if they have the same UnitID). 0=Reordering based on UnitID is supported.
4:0	Reserved.

D18F0x[F0,D0,B0,90] Link Base Channel Buffer Count

Read-write; Reset-applied.

Table 92: Register Mapping for D18F0x[F0,D0,B0,90]

Register	Function
D18F0x90	ONION Link
D18F0x[F0,D0,B0]	Reserved

D18F0x[F0,D0,B0,90] and D18F0x[F4,D4,B4,94] specify the *hard-allocated* link flow-control buffer counts in each virtual channel available to the transmitter at the other end of the link; it also provides the *free buffers* that may be used by any of the virtual channels, as needed. Base channel buffers are specified in D18F0x[F0,D0,B0,90]; isochronous buffer counts (if in IFCM) are specified in D18F0x[F4,D4,B4,94]. For all fields that specify buffer counts in D18F0x[F0,D0,B0,90] and D18F0x[F4,D4,B4,94]; If the link is ganged, then the number of buffers allocated is 2 times the value of the field; If the link is unganged, then the number of buffers allocated is the value of the field.

The cold or warm reset value is determined by whether the link initializes, whether the link is IO/coherent, whether the link is ganged/unganged, and whether the settings are locked by LockBc. Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary

even if the values are unchanged from the default values.

The hard-allocated buffer counts are transmitted to the device at the other end of the link in buffer release messages after link initialization. The remaining buffers are held in the free list (specified by FreeData and FreeCmd) used to optimize buffer usage. When a transaction is received, if a free-list buffer is available, it is used for storage instead of one of the hard allocated buffers; as a result, a buffer release (for one of the hard allocated buffers used by the incoming request) can be immediately sent back to the device at the other end of the link without waiting for the transaction to be routed beyond the flow-control buffers.

Table 93: Link Buffer Definitions

Term	Definition
LpbSize	Link Packet Command Buffer size. LpbSize = 48.
LpbdSize	Link Packet Data Buffer size. LpbdSize = 32.
LcsSize	Link Command Scheduler size. LcsSize = 48.

Buffer allocation rules:

- The total number of command buffers allocated in the base and isochronous registers of a link cannot exceed LpbSize:
 - (D18F0x[90][NpReqCmd] + D18F0x[90][PReq] + D18F0x[90][RspCmd] + D18F0x[90][ProbeCmd] + D18F0x[90][FreeCmd] + D18F0x[94][IsocNpReqCmd] + D18F0x[94][IsocPReq] + D18F0x[94][IsocRspCmd]) <= LpbSize.
- The total number of data buffers allocated in the base and isochronous registers of a link cannot exceed LpbdSize:
 - (D18F0x[90][NpReqData] + D18F0x[90][RspData] + D18F0x[90][PReq] + D18F0x[90][FreeData] + D18F0x[94][IsocPReq] + D18F0x[94][IsocNpReqData] + D18F0x[94][IsocRspData]) <= LpbdSize.
- The total number of hard allocated command buffers cannot exceed LcsSize.
 - (D18F0x[90][ProbeCmd] + D18F0x[90][RspCmd] + D18F0x[90][PReq] + D18F0x[90][NpReqCmd] + 18F0x[94][IsocRspCmd] + 18F0x[94][IsocPReq] + 18F0x[94][IsocNpReqCmd]) <= LcsSize.
- BIOS must set up non-zero counts (and adjust the base channel counts accordingly) prior to enabling IFCM.
- If an IOMMU is present in the system, D18F0x[F4,D4,B4,94][IsocNpReqCmd] must be non-zero for all enabled links.
- If an IOMMU is present in the system, D18F0x[E4,C4,A4,84][IsocEn] must be enabled for the ONION link.

Bits	Description
31	**LockBc: lock buffer count register**. Cold reset: 0. BIOS: 1. 1=The buffer count registers, D18F0x[F0,D0,B0,90] and D18F0x[F4,D4,B4,94] are locked such that warm resets do not place the registers back to their default value. Setting this bit does not prevent the buffer counts from being updated after a warm reset based on the value of the buffer counts before the warm reset. 0=Upon warm reset, the buffer count registers return to their default value after the link initializes regardless of the value before the warm reset.
30	**PReq[3]: posted request command and data buffer count [3]**. IF (LockBc) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. BIOS: IF (REG == D18F0x90) THEN 1. ELSE 0. ENDIF. See: PReq[2:0].
29:28	**NpReqData[3:2]: non-posted request data buffer count [3:2]**. IF (LockBc) THEN Cold reset: 11b. ELSE Reset: 00b. ENDIF. BIOS: IF (REG == D18F0x90) THEN 00b. ELSE 11b. ENDIF. See: NpReqData[1:0].
27:25	**FreeData: free data buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. BIOS: IF (REG == D18F0x90) THEN 2. ELSE 1. ENDIF.
24:20	**FreeCmd: free command buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF.
19:18	**RspData: response data buffer count**. IF (LockBc) THEN Cold reset: 1. ELSE Reset: 1. ENDIF. BIOS: IF (REG == D18F0x90) THEN 1. ELSE 0. ENDIF.
17:16	**NpReqData[1:0]: non-posted request data buffer count [1:0]**. NpReqData[3:0] = {NpReqData[3:2], NpReqData[1:0]}. IF (LockBc) THEN Cold reset: 01b. ELSE Reset: 01b. ENDIF. BIOS: IF (REG == D18F0x90) THEN 01b. ELSE 11b. ENDIF.
15:12	**ProbeCmd: probe command buffer count**. IF (LockBc) THEN Cold reset: 0h. ELSE Reset: 0h. ENDIF. BIOS: 0h.
11:8	**RspCmd: response command buffer count**. IF (LockBc) THEN Cold reset: 1h. ELSE Reset: 1h. ENDIF. BIOS: IF (REG == D18F0x90) THEN 2h. ELSE 0h. ENDIF.
7:5	**PReq[2:0]: posted request command and data buffer count [2:0]**. PReq[3:0] = {PReq[3], PReq[2:0]}. Specifies the number of posted command and posted data buffers allocated. IF (LockBc) THEN Cold reset: 110b. ELSE Reset: 110b. ENDIF. BIOS: IF (REG == D18F0x90) THEN 011b. ELSE 000b. ENDIF.
4:0	**NpReqCmd: non-posted request command buffer count**. IF (LockBc) THEN Cold reset: 09h. ELSE Reset: 09h. ENDIF. BIOS: IF (REG == D18F0x90) THEN 0Ah. ELSE 18h. ENDIF.

D18F0x[F4,D4,B4,94] Link Isochronous Channel Buffer Count

Read-write; Reset-applied. See D18F0x[F0,D0,B0,90].

Table 94: Register Mapping for D18F0x[F4,D4,B4,94]

Register	Function
D18F0x94	ONION Link
D18F0x[F4,D4,B4]	Reserved

Bits	Description
31:29	Reserved.
28:27	**IsocRspData: isochronous response data buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. BIOS: 0.
26:25	**IsocNpReqData: isochronous non-posted request data buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. BIOS: IF (REG == D18F0x94) THEN 1. ELSE 0. ENDIF.
24:22	**IsocRspCmd: isochronous response command buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. BIOS: 0.
21:19	**IsocPReq: isochronous posted request command and data buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. This specifies the number of isochronous posted command and posted data buffers allocated. BIOS: 0.
18:16	**IsocNpReqCmd: isochronous non-posted request command buffer count**. IF (D18F0x[F0,D0,B0,90][LockBc]) THEN Cold reset: 0. ELSE Reset: 0. ENDIF. BIOS: IF (REG == D18F0x94) THEN 1. ELSE 0. ENDIF.
15:8	**SecBusNum: secondary bus number**. Reset: 0. Specifies the configuration-space bus number of the IO link. When configured as a coherent link, this register has no meaning. This field should match the corresponding D18F1x[1DC:1D0,EC:E0][BusNumBase], unless D18F1x[1DC:1D0,EC:E0][DevCmpEn] == 1, in which case this field should be 00h).
7:0	Reserved.

D18F0x[F8,D8,B8,98] Link Type

Table 95: Register Mapping for D18F0x[F8,D8,B8,98]

Register	Function
D18F0x98	ONION Link
D18F0x[F8,D8,B8]	Reserved

Bits	Description
31:6	Reserved.
5	**PciEligible**. Read-only. Reset: 1.
4:0	Reserved.

D18F0x[11C,118,114,110] Link Clumping Enable

Reset: 0000_0002h. D18F0x[11C,118,114,110] are associated with the whole link if it is ganged or sublink 0 if it is unganged; If the node does not support a link, then the corresponding register addresses become reserved.

Table 96: Register Mapping for D18F0x[11C,118,114,110]

Register	Function
D18F0x110	ONION Link
D18F0x11[C:4]	Reserved

These registers specify how UnitIDs of upstream non-posted requests may be clumped per the link specification. The processor does not clump requests that it generates in the downstream direction.

Bits	Description
31:1	**ClumpEn**. Read-write. Each bit of this register corresponds to a link UnitID number (e.g., bit[2] corresponds to UnitID 02h, etc). 1=The specified UnitID is ordered in the same group as the specified UnitID - 1. For example, if this register is programmed to 0000_00C0h, then UnitIDs 7h, 6h, and 5h are all ordered as if they are part of the same UnitID. This is used to allow more than 32 tags to be assigned to a single stream for the purposes of ordering.
0	Reserved.

D18F0x150 Link Global Retry Control

Cold reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D18F0x168 Extended Link Transaction Control

Read-write.

Bits	Description
31:21	Reserved.
20	**XcsSecPickerDstNcHt**. Reset: 0. BIOS: 1.
19:0	Reserved.

D18F0x16C Link Global Extended Control

Reset: 0000_0000h.

Bits	Description
31:0	Reserved.

D18F0x[18C:170] Link Extended Control

These registers provide control for each link. They are mapped to the links as follows:

Table 97: Register Mapping for D18F0x[18C:170]

Register	Function
D18F0x170	ONION Link
D18F0x1[8C:74]	Reserved

Reset: 0000_0001h.

Bits	Description
31:0	Reserved.

D18F0x1A0 Link Initialization Status

Table 98: Onion Definitions

Term	Definition
OnionPlus	OnionPlus link detected. OnionPlus = (D18F0x1A0[OnionPlusCap]).

Bits	Description
31	**InitStatusValid: initialization status valid**. Read-only; Updated-by-hardware. Reset: 0. 1=Indicates that the rest of the information in this register is valid for all links; each link is either not connected or the initialization is complete.
30:28	Reserved.
27:24	**OnionPlusCap**.Read-only; Updated-by-hardware. Reset: 0. 1=OnionPlus capable link detected. Bit — Description [0] — Link 0 [1] — Link 1 [2:3] — Reserved
23:4	Reserved.
3:2	**InitComplete1: initialization complete for link 1**. See: InitComplete0.
1:0	**InitComplete0: initialization complete for link 0**. Read-only; Updated-by-hardware. Reset: 00b. Bits — Description 00b — Internal northbridge link has not completed initialization. 10b-01b — Reserved. 11b — Internal northbridge link has completed initialization.

D18F0x1DC Core Enable

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.

15:1	**CpuEn: core enable**. Read-write. This field is used to enable each of the cores after a reset. 1=Enable the core to start fetching and executing code from the boot vector. The most significant bit N is indicated by CpuCoreNum, as defined in section 2.4.4 [Processor Cores and Downcoring]. All bits greater than N are reserved. Bit Description [0] Core 1 enable. [1] Core 2 enable. [15:2] Reserved.
0	Reserved.

D18F0x1F8 ONION3 Link Status and Control Register

Bits	Description
31:18	Reserved.
17	**On3LnkFail1**. Cold reset: 0. Read; Set-by-hardware; Write-1-to-clear. Set when a Sync flood is received to indicate ONION3 Link 1 failed.
16	**On3LnkFail0**. Cold reset: 0. Read; Set-by-hardware; Write-1-to-clear. Set when a Sync flood is received to indicate ONION3 Link 0 failed.
15:10	Reserved.
9	**On3LnkInitComp1**. Read-only. Reset: 0. Indicates ONION3 Link 1 initialization is complete.
8	**On3LnkInitComp0**. Read-only. Reset: 0. Indicates ONION3 Link 0 initialization is complete.
7:6	Reserved.
5	**On3LnkConn1**. Read-only. Reset: 1. Indicates ONION3 Link 1 is connected.
4	**On3LnkConn0**. Read-only. Reset: 1. Indicates ONION3 Link 0 is connected.
3:2	Reserved.
1	**On3LnkCap1**. Read-only. Reset: 1. Indicates ONION3 Link 1 instance exists.
0	**On3LnkCap0**. Read-only. Reset: 1. Indicates ONION3 Link 0 instance exists.

D18F0x20[4:0] ONION3 Upstream Base Channel Buffer Count

Table 99: Register Mapping for D18F0x20[4:0]

Register	Function
D18F0x200	Link 0
D18F0x204	Link 1

Bits	Description
31	**BufCntLock**. Read-write. Reset: 0. 1=The current buffer count register is locked such that warm resets do not place the registers back to their default value. 0=Upon warm reset, the buffer count registers return to their default value after the link initializes regardless of the value before the warm reset.
30:22	Reserved.

21:16	**On3ReqDatBufCnt**. Read-write. Reset: 20h. Specifies the ONION3 Request Data buffer count.
15:14	Reserved.
13:8	**On3RspBufCnt**. Read-write. Reset: 3Fh. Specifies the ONION3 Response buffer count.
7:6	Reserved.
5:0	**On3ReqBufCnt**. Read-write. Reset: 30h. Specifies the ONION3 Request buffer count.

3.11 Device 18h Function 1 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D18F1x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 15B1h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D18F1x08 Class Code/Revision ID

Bits	Description
31:8	**ClassCode**. Read-only. Value: 06_0000h. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only. Value: 00h. Processor revision. 00h=A0.

D18F1x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:0	**HeaderTypeReg**. Read-only. These bits are fixed at their default values. The header type field indicates that there are multiple functions present in this device.

D18F1x[17C:140,7C:40] DRAM Base/Limit

The following sets of registers specify the DRAM address ranges:

Table 100: Register Mapping for D18F1x[17C:140,7C:40]

Function	Base Low	Limit Low	Base High	Limit High
Range 0	D18F1x40	D18F1x44	D18F1x140	D18F1x144
Reserved	D18F1x48, D18F1x[7:5][8,0]	D18F1x4C, D18F1x[7:5][C,4]	D18F1x148, D18F1x1[7:5][8,0]	D18F1x14C, D18F1x1[7:5][C,4]

Transaction addresses that are within the specified base/limit range are routed to the DstNode. See 2.8.2 [NB Routing].

DRAM mapping rules:
F1x0XX registers provide the low address bits. F1x1XX registers provide the high address bits.

- Transaction addresses are within the defined range if:
 {DramBase[47:24], 00_0000h} <= address[47:0] <= {DramLimit[47:24], FF_FFFFh}.
- DRAM regions must not overlap each other.
- Accesses to addresses that map to both DRAM, as specified by the DRAM base and limit registers (F1x[1, 0][7C:40]), and MMIO, as specified by D18F1x[2CC:2A0,1CC:180,BC:80], are routed to MMIO only.

- Programming of the DRAM address maps must be consistent with the Memory-Type Range Registers (MTRRs) and the top of memory registers, MSRC001_001A and MSRC001_001D. CPU accesses only hit within the DRAM address maps if the corresponding MTRR is of type DRAM. Accesses from IO links are routed based on the DRAM base and limit registers (F1x[1, 0][7C:40]) only.
- The appropriate RE or WE bit(s) must be set. When initializing a base/limit pair, the BIOS must write the [limit] register before either the RE or WE bit is set. When changing a base/limit pair that is already enabled, the BIOS should clear RE and WE before changing the address range.
- See 2.8.2.1.1 [DRAM and MMIO Memory Space].

When memory hoisting is enabled in a node via D18F1x2[1,0][8,0][LgcyMmioHoleEn], the corresponding BaseAddr/LimitAddr should be configured to account for the memory hoisted above the hole. See 2.9.12 [Memory Hoisting].

D18F1x[7:4][8,0] DRAM Base Low

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0000_0000h.

Table 101: Register Mapping for D18F1x[7:4][8,0]

Register	Function
D18F1x40	Range 0
D18F1x48	Reserved
D18F1x[7:5][8,0]	Reserved

Bits	Description
31:16	**DramBase[39:24]: DRAM base address register bits[39:24]**. DramBase[47:24] = {D18F1x1[7:4][8,0][DramBase[47:40]], D18F1x[7:4][8,0][DramBase[39:24]]}.
15:2	Reserved.
1	**WE: write enable**. 1=Writes to this address range are enabled.
0	**RE: read enable**. 1=Reads to this address range are enabled.

D18F1x1[7:4][8,0] DRAM Base High

Table 102: Register Mapping for D18F1x1[7:4][8,0]

Register	Function
D18F1x140	Range 0
D18F1x148	Reserved
D18F1x1[7:5][8,0]	Reserved

Bits	Description
31:8	Reserved.
7:0	**DramBase[47:40]: DRAM base address register bits[47:40]**. See: D18F1x[7:4][8,0][DramBase[39:24]].

D18F1x[7:4][C,4] DRAM Limit Low

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF.

Table 103: Register Mapping for D18F1x[7:4][C,4]

Register	Function
D18F1x44	Range 0
D18F1x4C	Reserved
D18F1x[7:5][C,4]	Reserved

Bits	Description
31:16	**DramLimit[39:24]: DRAM limit address register bits[39:24]**. Reset: FCFFh. DramLimit[47:24] = {D18F1x1[7:4][C,4][DramLimit[47:40]], D18F1x[7:4][C,4][DramLimit[39:24]]}.
15:3	Reserved.
2:0	**DstNode: destination Node ID**. Reset: 000b. Specifies the node that a packet is routed to if it is within the address range.

D18F1x1[7:4][C,4] DRAM Limit High

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF.

Table 104: Register Mapping for D18F1x1[7:4][C,4]

Register	Function
D18F1x144	Range 0
D18F1x14C	Reserved
D18F1x1[7:5][C,4]	Reserved

Bits	Description
31:8	Reserved.
7:0	**DramLimit[47:40]: DRAM limit address register bits[47:40]**. Reset: 00h. See D18F1x[7:4][C,4][DramLimit[39:24]].

D18F1x[2CC:2A0,1CC:180,BC:80] MMIO Base/Limit

These registers, The memory mapped IO base and limit registers D18F1x[2CC:2A0,1CC:180,BC:80] specify the mapping from memory addresses to the corresponding node and IO link for MMIO transactions. Address ranges are specified by upto 16 sets of base/limit registers.

Table 105: Register Mapping for D18F1x[2CC:2A0,1CC:180,BC:80]

Function	MMIO Base Low	MMIO Limit Low	MMIO Base/Limit High
Range 0	D18F1x80	D18F1x84	D18F1x180
Range 1	D18F1x88	D18F1x8C	D18F1x184
Range 2	D18F1x90	D18F1x94	D18F1x188
Range 3	D18F1x98	D18F1x9C	D18F1x18C

Table 105: Register Mapping for D18F1x[2CC:2A0,1CC:180,BC:80] (Continued)

Range 4	D18F1xA0	D18F1xA4	D18F1x190
Range 5	D18F1xA8	D18F1xAC	D18F1x194
Range 6	D18F1xB0	D18F1xB4	D18F1x198
Range 7	D18F1xB8	D18F1xBC	D18F1x19C
Range 8	D18F1x1A0	D18F1x1A4	D18F1x1C0
Range 9	D18F1x1A8	D18F1x1AC	D18F1x1C4
Range 10	D18F1x1B0	D18F1x1B4	D18F1x1C8
Range 11	D18F1x1B8	D18F1x1BC	D18F1x1CC
Reserved	D18F1x2[B8,B0,A8,A0]	D18F1x2[BC,B4,AC,A4]	D18F1x2[CC,C8,C4,C0]

Transaction addresses that are within the specified base/limit range are routed to the node specified by DstNode and the link specified by DstLink. See 2.8.2 [NB Routing].

MMIO mapping rules:
- Transaction addresses are within the defined range if: {MMIOBase[47:16], 0000h} <= address[47:0] <= {MMIOLimit[47:16], FFFFh}.
- MMIO regions must not overlap each other.
- Accesses to addresses that map to both DRAM, as specified by the DRAM base and limit registers (see D18F1x[17C:140,7C:40]), and MMIO, as specified by the memory mapped IO base and limit registers (F1x[BC:80]), are routed to MMIO only.
- Programming of the MMIO address maps must be consistent with the Memory-Type Range Registers (MTRRs) and the top of memory registers, MSRC001_001A and MSRC001_001D. CPU accesses only hit within the MMIO address maps if the corresponding MTRR is of type IO. Accesses from IO links are routed based on D18F1x[2CC:2A0,1CC:180,BC:80].
- The appropriate RE or WE bit(s) must be set. When initializing a base/limit pair, the BIOS must write the limit register before either the RE or WE bit is set. When changing a base/limit pair that is already enabled, the BIOS should clear RE and WE before changing the address range.
- Scenarios in which the address space of multiple MMIO ranges target the same IO device is supported.
- See 2.8.2.1.1 [DRAM and MMIO Memory Space].

D18F1x[2B:1A,B:8][8,0] MMIO Base Low

Table 106: Register Mapping for D18F1x[2B:1A,B:8][8,0]

Register	Function
D18F1x80	Range 0
D18F1x88	Range 1
D18F1x90	Range 2
D18F1x98	Range 3
D18F1xA0	Range 4
D18F1xA8	Range 5
D18F1xB0	Range 6
D18F1xB8	Range 7
D18F1x1A0	Range 8
D18F1x1A8	Range 9

Table 106: Register Mapping for D18F1x[2B:1A,B:8][8,0] (Continued)

D18F1x1B0	Range 10
D18F1x1B8	Range 11
D18F1x2[B:A][8,0]	Reserved

Bits	Description
31:8	**MMIOBase[39:16]: MMIO base address register bits[39:16]**. Read-write. Reset: 0. MMIOBase[47:16] = {D18F1x[2CC:2C0,1CC:1C0,19C:180][MMIOBase[47:40]], MMIO-Base[39:16]}.
7:4	Reserved.
3	**Lock**. Read-write. Reset: 0. 1=The memory mapped IO base and limit registers (D18F1x[2CC:2A0,1CC:180,BC:80]) are read-only (including this bit). WE or RE in this register must be set in order for this to take effect.
2	Reserved.
1	**WE: write enable**. Read-write. Reset: 0. 1=Writes to this address range are enabled.
0	**RE: read enable**. Read-write. Reset: 0. 1=Reads to this address range are enabled.

D18F1x[2B:1A,B:8][C,4] MMIO Limit Low

Table 107: Register Mapping for D18F1x[2B:1A,B:8][C,4]

Register	Function
D18F1x84	Range 0
D18F1x8C	Range 1
D18F1x94	Range 2
D18F1x9C	Range 3
D18F1xA4	Range 4
D18F1xAC	Range 5
D18F1xB4	Range 6
D18F1xBC	Range 7
D18F1x1A4	Range 8
D18F1x1AC	Range 9
D18F1x1B4	Range 10
D18F1x1BC	Range 11
D18F1x2[BC,B4,AC,A4]	Reserved

Bits	Description
31:8	**MMIOLimit[39:16]: MMIO limit address register bits[39:16]**. Read-write. Reset: 0. MMIOLimit[47:16] = {D18F1x[2CC:2C0,1CC:1C0,19C:180][MMIOLimit[47:40]], MMIOLimit[39:16]}.

7	**NP: non-posted**. Read-write. Reset: 0. 1=CPU write requests to this MMIO range are passed through the non-posted channel. This may be used to force writes to be non-posted for MMIO regions which map to the legacy ISA/LPC bus, or in conjunction with D18F0x68[DsNpReqLmt] in order to allow downstream CPU requests to be counted and thereby limited to a specified number. This latter use of the NP bit may be used to avoid loop deadlock scenarios in systems that implement a region in an IO device that reflects downstream accesses back upstream. See the link summary of deadlock scenarios for more information. 0=CPU writes to this MMIO range use the posted channel. This bit does not affect requests that come from IO links (the virtual channel of the request is specified by the IO request). If two MMIO ranges target the same IO device and the NP bit is set differently in both ranges, unexpected transaction ordering effects are possible. In particular, using PCI- and IO-link-defined producer-consumer semantics, if a producer (i.e., the processor) writes data using a non-posted MMIO range followed by a flag to a posted MMIO range, then it is possible for the device to see the flag updated before the data is updated.
6	**DstSubLink: destination sublink**. Read-write. Reset: 0. When a link is unganged, this bit specifies the destination sublink of the link specified by the memory mapped IO base and limit registers F1x[BC:80][DstLink]. 0=The destination link is sublink 0. 1=The destination link is sublink 1. If the link is ganged, then this bit must be low.
5:4	**DstLink: destination link ID**. Read-write. Reset: 0. For transactions within this MMIO range, this field specifies the destination IO link number of the destination node. Bits Description 00b Link 0 01b Link 1 10b Link 2 11b Link 3
3	Reserved.
2:0	**DstNode: destination node ID bits**. Read-write. Reset: 0. For transactions within this MMIO range, this field specifies the destination node ID.

D18F1x[2CC:2C0,1CC:1C0,19C:180] MMIO Base/Limit High

Table 108: Register Mapping for D18F1x[2CC:2C0,1CC:1C0,19C:180]

Register	Function
D18F1x180	Range 0
D18F1x184	Range 1
D18F1x188	Range 2
D18F1x18C	Range 3
D18F1x190	Range 4
D18F1x194	Range 5
D18F1x198	Range 6
D18F1x19C	Range 7
D18F1x1C0	Range 8
D18F1x1C4	Range 9

Table 108: Register Mapping for D18F1x[2CC:2C0,1CC:1C0,19C:180] (Continued)

D18F1x1C8	Range 10
D18F1x1CC	Range 11
D18F1x2[CC:C0]	Reserved

Bits	Description
31:24	Reserved.
23:16	**MMIOLimit[47:40]: MMIO limit address register bits[47:40]**. See: D18F1x[2B:1A,B:8][C,4][MMIOLimit[39:16]].
15:8	Reserved.
7:0	**MMIOBase[47:40]: MMIO base address register bits[47:40]**. See: D18F1x[2B:1A,B:8][8,0][MMIOBase[39:16]].

D18F1x[DC:C0] IO-Space Base/Limit

The IO-space base and limit registers, D18F1x[DC:C0], specify the mapping from IO addresses to the corresponding node and IO link for transactions resulting from x86-defined IN and OUT instructions. IO address ranges are specified by upto 8 sets of base/limit registers. The first set is F1xC0 and F1xC4, the second set is F1xC8 and F1xCC, and so forth. Transaction addresses that are within the specified base/limit range are routed to the node specified by DstNode and the link specified by DstLink. See 2.8.2 [NB Routing].

IO mapping rules:

- IO-space transaction addresses are within the defined range if: {IOBase[24:12], 000h} <= address <= {IOLimit[24:12], FFFh} and as specified by the IE bit; or if the address is in the range specified by the VE bits.
- IO regions must not overlap each other.
- The appropriate RE or WE bit(s) must be set.
- See 2.8.2.1.2 [IO Space].

D18F1x[1F:1E,D:C][8,0] IO-Space Base

Table 109: Register Mapping for D18F1x[1F:1E,D:C][8,0]

Register	Function
D18F1xC0	Range 0
D18F1xC8	Range 1
D18F1xD0	Range 2
D18F1xD8	Range 3
D18F1x1[F:E][8,0]	Reserved

Bits	Description
31:25	Reserved.
24:12	**IOBase[24:12]: IO base address register bits[24:12]**. Read-write. Reset: 0.
11:6	Reserved.

5	**IE: ISA enable**. Read-write. Reset: 0. 1=The IO-space address window is limited to the first 256 B of each 1 KB block specified; this only applies to the first 64 KB of IO space. 0=The PCI IO window is not limited in this way.
4	**VE: VGA enable**. Read-write. Reset: 0. 1=Include IO-space transactions targeting the VGA-compatible address space within the IO-space window of this base/limit pair. These include IO accesses in which address bits[9:0] range from 3B0h to 3BBh or 3C0h to 3DFh (address bits[15:10] are not decoded); this only applies to the first 64 KB of IO space; i.e., address bits[24:16] must be low). 0=IO-space transactions targeting VGA-compatible address ranges are not added to the IO-space window. This bit should only ever be set in one register. The MMIO range associated with the VGA enable bit in the PCI specification is not included in the VE bit definition; to map this range to an IO link, see D18F1xF4 [VGA Enable]. When D18F1xF4[VE] is set, the state of this bit is ignored.
3:2	Reserved.
1	**WE: write enable**. Read-write. Reset: 0. 1=Writes to this IO-space address range are enabled.
0	**RE: read enable**. Read-write. Reset: 0. 1=Reads to this IO-space address range are enabled.

D18F1x[1F:1E,D:C][C,4] IO-Space Limit

Table 110: Register Mapping for D18F1x[1F:1E,D:C][C,4]

Register	Function
D18F1xC4	Range 0
D18F1xCC	Range 1
D18F1xD4	Range 2
D18F1xDC	Range 3
D18F1x1[F:E][C,4]	Reserved

Bits	Description
31:25	Reserved.
24:12	**IOLimit[24:12]: IO limit address register bits[24:12]**. Read-write. Reset: 0.
11:7	Reserved.
6	**DstSubLink: destination sublink**. Read-write. Reset: 0. When a link is unganged, this bit specifies the destination sublink of the link specified by F1x[DC:C0][DstLink]. 0=The destination link is sublink 0. 1=The destination link is sublink 1. If the link is ganged, then this bit must be low.
5:4	**DstLink: destination link ID**. Read-write. Reset: 0. For transactions within this IO-space range, this field specifies the destination IO link number of the destination node. Bits — Description 00b — Link 0 01b — Link 1 10b — Link 2 11b — Link 3
3	Reserved.
2:0	**DstNode: destination node ID bits**. Read-write. Reset: 0. For transactions within this IO-space range, this field specifies the destination node ID.

D18F1x[1DC:1D0,EC:E0] Configuration Map

D18F1x[1DC:1D0,EC:E0] specify the mapping from configuration address to the corresponding node and IO link. Configuration address ranges are specified by upto 8 pairs of base/limit registers. Transaction addresses that are within the specified base/limit range are routed to the node specified by DstNode and the link specified by DstLink. See 2.8.2 [NB Routing].

Table 111: Register Mapping for D18F1x[1DC:1D0,EC:E0]

Register	Function
D18F1xE0	Range 0
D18F1xE4	Range 1
D18F1xE8	Range 2
D18F1xEC	Range 3
D18F1x1[DC:D0]	Reserved

Configuration space mapping rules:

- Configuration addresses (to"BusNo" and "Device" as specified by IOCF8 [IO-Space Configuration Address] in the case of IO accesses or 2.7 [Configuration Space] in the case of MMIO accesses) are within the defined range if:
 ({BusNumBase[7:0]} <= BusNo <= {BusNumLimit[7:0]}) & (DevCmpEn==0); or
 ({BusNumBase[4:0]} <= Device <= {BusNumLimit[4:0]}) & (DevCmpEn==1) & (BusNo == 00h).
- Configuration regions must not overlap each other.
- The appropriate RE or WE bit(s) must be set.
- See 2.8.2.1.3 [Configuration Space].

Bits	Description
31:24	**BusNumLimit[7:0]: bus number limit bits[7:0]**. Read-write. Reset: 0.
23:16	**BusNumBase[7:0]: bus number base bits[7:0]**. Read-write. Reset: 0.
15:3	Reserved.
2	**DevCmpEn: device number compare mode enable**. Read-write. Reset: 0. 1=A device number range rather than a bus number range is used to specify the configuration-space window (see above).
1	**WE: write enable**. Read-write. Reset: 0. 1=Writes to this configuration-space address range are enabled.
0	**RE: read enable**. Read-write. Reset: 0. 1=Reads to this configuration-space address range are enabled.

D18F1xF0 DRAM Hole Address

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Same-for-all. Reset: 0000_0000h. See 2.9.12 [Memory Hoisting].

Bits	Description
31:24	**DramHoleBase[31:24]: DRAM hole base address**. Specifies the base address of the IO hole, below the 4GB address level, that is used in memory hoisting. Normally, DramHoleBase >= MSRC001_001A[TOM[31:24]].
23:16	Reserved.

15:7	**DramHoleOffset[31:23]: DRAM hole offset address**. When D18F1x2[1,0][8,0][LgcyMmioHoleEn] == 1, this offset is subtracted from the physical address of certain accesses in forming the normalized address.
6:2	Reserved.
1	**DramMemHoistValid: dram memory hoist valid**. 1=Memory hoisting for the address range is enabled. 0=Memory hoisting is not enabled. This bit should be set if any D18F1x2[1,0][8,0][LgcyMmioHoleEn] == 1 or DramHoleBase != 0.
0	**DramHoleValid: dram hole valid**. 1=Memory hoisting is enabled in the node. 0=Memory hoisting is not enabled. This bit should be set in the node that owns the DRAM address space that is hoisted above the 4 GB address level. See DramHoleBase.

D18F1xF4 VGA Enable

Reset: 0000_0000h.

Bits	Description
31:15	Reserved.
14	**DstSubLink: destination sublink**. Read-write. When a link is unganged, this bit specifies the destination sublink of the link specified by D18F1xF4[DstLink]. 0=The destination link is sublink 0. 1=The destination link is sublink 1. If the link is ganged, then this bit must be low.
13:12	**DstLink: destination link ID**. Read-write. For transactions within the D18F1xF4[VE]-defined ranges, this field specifies the destination IO link number of the destination node. Bits — Description 00b — Link 0 01b — Link 1 10b — Link 2 11b — Link 3
11:7	Reserved.
6:4	**DstNode: destination node ID**. Read-write. For transactions within the D18F1xF4[VE]-defined range, this field specifies the destination node ID.
3	**Lock**. Read-write. 1=All the bits in this register (D18F1xF4) are read-only (including this bit).
2	Reserved.
1	**NP: non-posted**. Read-write. 1=CPU write requests to the D18F1xF4[VE]-defined MMIO range are passed through the non-posted channel. 0=CPU writes may be posted.
0	**VE: VGA enable**. Read-write. 1=Transactions targeting the VGA-compatible address space are routed and controlled as specified by this register. The VGA-compatible address space is: (1) the MMIO range A_0000h through B_FFFFh; (2) IO-space accesses in which address bits[9:0] range from 3B0h to 3BBh or 3C0h to 3DFh (address bits[15:10] are not decoded; this only applies to the first 64 KB of IO space; i.e., address bits[24:16] must be low). 0=Transactions targeting the VGA-compatible address space are not affected by the state of this register. When this bit is set, the state of D18F1xF4[VE] is ignored.

D18F1x10C DCT Configuration Select

Reset: 0000_0000h.

Bits	Description
31:10	Unused.
9:8	Reserved.
7	**DctCfgBcEn: Dct config write broadcast enable**. Read-write. 1=For a logical CSR write, all enabled DCTs receive the config data. 0=Only the DCT specified with DctCfgSel receives the config data of a config write. For a logical CSR read, DctCfgBcEn=x and the DCT with DctCfgSel supplies the read data.
6:3	Unused.
2:0	**DctCfgSel: DRAM controller configuration select**. Read-write. Specifies DCT controller to which accesses are routed. See 2.9.3 [DCT Configuration Registers]. Bits: Description 000b: DCT 0 111b-001b: Reserved

D18F1x120 DRAM Base System Address

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Cold reset: 0000_0000h.
D18F1x120 and D18F1x124 are required to specify the base and limit system address range of the DRAM connected to the local node.
DRAM accesses to the local node with physical address Addr[47:0] that are within the following range are directed to the DCTs:

{DramBaseAddr[47:27], 000_0000h} <= Addr[47:0] <= {DramLimitAddr[47:27], 7FF_FFFFh};

DRAM accesses to the local node that are outside of this range are master aborted.

The address of the DRAM transaction is normalized before passing it to the DCTs by subtracting DramBaseAddr.

Bits	Description
31:21	Reserved.
20:0	**DramBaseAddr[47:27]**.

D18F1x124 DRAM Limit System Address

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. See D18F1x120 [DRAM Base System Address].

Bits	Description
31:21	Reserved.
20:0	**DramLimitAddr[47:27]**. Cold reset: 1F_FFFFh.

D18F1x2[1C:00] DRAM Controller Base/Limit

The DRAM controller base and limit registers define a DRAM controller address range and specify the mapping of physical DRAM addresses to a DCT as selected by DctSel or DctIntLvEn. The following base/limit register pairs specify the address ranges:

Table 112: Register Mapping for D18F1x2[1C:00]

Function	Base Address	Limit Address
Range 0	D18F1x200	D18F1x204
Range 1	D18F1x208	D18F1x20C
Reserved	D18F1x21[8,0]	D18F1x21[C,4]

BIOS should observe the following DCT configuration requirements:

- DRAM addresses are within the defined range if:
 {DctBaseAddr[47:27], 000b, 00_0000h} <= address[47:0] <= {DctLimitAddr[47:27], 111b, FF_FFFFh}.
- DCT base/limit address ranges must not overlap each other.
- A maximum of two address ranges may be mapped to a single DCT.

Hoisting. When memory hoisting is enabled viaLegacyMmioHoleEn, the corresponding DctBaseAddr/DctLimitAddr should be configured to account for the memory hoisted above the hole. A contiguous memory hole should only be mapped by one DctBaseAddr/DctLimitAddr pair. See 2.9.12 [Memory Hoisting].

D18F1x2[1,0][8,0] DRAM Controller Base

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0000_0000h.

Table 113: Register Mapping for D18F1x2[1,0][8,0]

Register	Function
D18F1x200	Range 0
D18F1x208	Range 1
D18F1x21[8,0]	Reserved

Bits	Description
31:11	**DctBaseAddr[47:27]: DRAM controller base address [47:27]**. Read-write. Specifies the base physical address bits for this address range.
10:7	Reserved.
6:4	**DctSel: DRAM controller select**. Read-write. Specifies the DCT mapped to this address range. Bits — Definition 000b — DCT 0 111b-010b — Reserved
3:2	Reserved.

1	**LgcyMmioHoleEn: legacy mmio hole enable**. Read-write. BIOS: See 2.9.12 [Memory Hoisting]. 1=Enable memory hoisting for this address range. BIOS sets this bit for an address range that spans the 4GB boundary and contains a hole for addresses used by MMIO. 0=Memory hoisting is not enabled.
0	**DctAddrVal: DRAM controller address valid**. Read-write. 1=Specifies this address range is valid and enabled. 0=This address range is not enabled.

D18F1x2[1,0][C,4] DRAM Controller Limit

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0000_0000h.

Table 114: Register Mapping for D18F1x2[1,0][C,4]

Register	Function
D18F1x204	Range 0
D18F1x20C	Range 1
D18F1x21[C,4]	Reserved

Bits	Description
31:11	**DctLimitAddr[47:27]: DRAM controller limit address bits [47:27]**. Read-write. Specifies the limit physical address bits for this address range.
10:0	Reserved.

3.12 Device 18h Function 2 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D18F2x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 15B2h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D18F2x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F2x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:0	**HeaderTypeReg**. Read-only. These bits are fixed at their default values. The header type field indicates that there multiple functions present in this device.

D18F2x[5C:40]_dct[0] DRAM CS Base Address

See 2.9.3 [DCT Configuration Registers].

These registers along with D18F2x[6C:60]_dct[0] [DRAM CS Mask], translate DRAM request addresses (to a DRAM controller) into DRAM chip selects. Supported DIMM sizes are specified in D18F2x80_dct[0] [DRAM Bank Address Mapping]. For more information on the DRAM controllers, see 2.9 [DRAM Controllers (DCTs)].

For each chip select, there is a DRAM CS Base Address register. For each CS pair there is a DRAM CS Mask Register. For each CS pair, an even CS must be populated if the odd CS is populated. If a chipselect is populated in the system it must be indicated to the DCT by setting one of the mutually exclusive {CSEnable, TestFail} configuration bits.

Table 115: DIMM, Chip Select, and Register Mapping

Base Address Registers	Mask Register	Logical DIMM	Chip Select[1]
D18F2x40_dct[x]	F2x60	0	MEMCS[x]_L[0]
D18F2x44_dct[x]			MEMCS[x]_L[1]

Table 115: DIMM, Chip Select, and Register Mapping (Continued)

Base Address Registers	Mask Register	Logical DIMM	Chip Select[1]
D18F2x48_dct[x]	F2x64	1	MEMCS[x]_L[2]
D18F2x4C_dct[x]			MEMCS[x]_L[3]
D18F2x50_dct[x]	F2x68	2	MEMCS[x]_L[4]
D18F2x54_dct[x]			MEMCS[x]_L[5]
D18F2x58_dct[x]	F2x6C	3	MEMCS[x]_L[6]
D18F2x5C_dct[x]			MEMCS[x]_L[7]
1. See 2.9.4 [DDR Pad to Processor Pin Mapping]			

The DRAM controller operates on the normalized physical address of the DRAM request. The normalized physical address includes all of the address bits that are supported by a DRAM controller. See 2.8 [Northbridge (NB)].

Each base address register specifies the starting normalized address of the block of memory associated with the chip select. Each mask register specifies the additional address bits that are consumed by the block of memory associated with the chip selects. If both chip selects of a DIMM are used, they must be the same size; in this case, a single mask register covers the address space consumed by both chip selects.

Lower-order address bits are provided in the base address and mask registers, as well. These allow memory to be interleaved between chip selects, such that contiguous physical addresses map to the same DRAM page of multiple chip selects. See 2.9.11.1 [Chip Select Interleaving]. The hardware supports the use of lower-order address bits to interleave chip selects if (1) the each chip select of the memory system spans the same amount of memory and (2) the number of chip selects of the memory system is a power of two.

BIOS is required to assign the largest DIMM chip-select range to the lowest normalized address of the DRAM controller. As addresses increase, the chip-select size is required to remain constant or decrease. This is necessary to keep DIMM chip-select banks on aligned address boundaries, regardless as to the amount of address space covered by each chip select.

For each normalized address for requests that enters a DRAM controller, a ChipSelect[i] is asserted if:

```
CSEnable[i] &
( {(InputAddr[39:27]   & ~AddrMask[i][39:27]),
        (InputAddr[21:11]   & ~AddrMask[i][21:11])} ==
       {(BaseAddr[i][39:27] & ~AddrMask[i][39:27]),
        (BaseAddr[i][21:11]  & ~AddrMask[i][21:11])} );
```

Bits	Description
31:19	**BaseAddr[39:27]: normalized physical base address bits[39:27]**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
18:16	Reserved.
15:5	**BaseAddr[21:11]: normalized physical base address bits[21:11]**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
4	Reserved.

3	**OnDimmMirror: on-DIMM mirroring (ODM) enabled**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 1=Address and bank bits are swapped by hardware for MRS commands sent to this chip select. This mode accounts for routing on the DIMM. Hardware bit swapping does not occur for commands sent via D18F2x7C_dct[0][SendMrsCmd] when D18F2x7C_dct[0][EnDramInit] == 0. This bit is expected to be set for the odd numbered rank of UDIMMs if SPD byte 63 indicates that address mapping is mirrored. The following bits are swapped when enabled for Ddr3Mode: • BA0 and BA1. • A3 and A4. • A5 and A6. • A7 and A8. The following bits are swapped when enabled for Ddr4Mode: • BA0 and BA1. • BG0 and BG1. • A3 and A4. • A5 and A6. • A7 and A8. • A11 and A13.
2	**TestFail: memory test failed**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Set by BIOS to indicate that a rank is present but has failed memory training or a memory consistency test, indicating that the memory is bad or unused.
1	Reserved.
0	**CSEnable: chip select enable**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.

D18F2x[6C:60]_dct[0] DRAM CS Mask

See 2.9.3 [DCT Configuration Registers] and D18F2x[5C:40]_dct[0].

Bits	Description
31:19	**AddrMask[39:27]: normalized physical address mask bits[39:27]**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
18:16	Reserved.
15:5	**AddrMask[21:11]: normalized physical address mask bits[21:11]**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
4:0	Reserved.

D18F2x78_dct[0] DRAM Control

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31	Reserved.

30	**DramSprLock**. Write-1-only. Reset: 0. BIOS: D18F2x118[LockDramCfg]. The following registers are read-only if DramSprLock == 1, otherwise the access type is specified by the register: • D18F2x[5C:40]_dct[0] [DRAM CS Base Address] • D18F2x[6C:60]_dct[0] [DRAM CS Mask] • D18F2x80_dct[0] [DRAM Bank Address Mapping] • D18F2x250_dct[0] [DRAM Loopback and Training Control]
29:28	Reserved.
27:20	Reserved.
19	Reserved.
18	**BgMirrorDis: bank group mirror dis**. Read-write. Reset: 0. 1=Disable swap of BG1 & BG0 for mirrored ranks. 0=Swap BG1 & BG0 for mirrored ranks. Reserved if !Ddr4Mode. BIOS sets this bit if 8-Bank x16 DRAMs are present and OnDimmMirror=1.
17	**AddrCmdTriEn: address command tristate enable**. Read-write. Reset: 0. BIOS: See 2.9.9.4. 1=Tristate the address, command, and bank buses when a Deselect command is issued.
16:11	Reserved.
10:8	**DramType: DRAM type**. Read-write. Reset: 7h. Specifies the type of DRAM devices which are connected to this DCT. Bits — Description 000b — DDR3 001b — Reserved 010b — DDR4 011b — Reserved 110b-100b — Reserved 111b — Undefined
7:5	Reserved.
4	**PtrInitReq: fifo pointer initialization request**. Read; Write-1-only; Cleared-when-done. Reset: 0. BIOS: See 2.9.9.7. 1=The DCT performs transmit and receive fifo pointer initialization. This bit is cleared by hardware after the initialization completes.
3	Reserved.
2	**GsyncDis: G-sync bus disable**. Read-write. Reset: 1. BIOS: See 2.9.9.4. 0=Enable the G-sync bus communication between phys and the PLL. 1=G-sync bus is masked.
1	Reserved.
0	**ChanVal: DRAM channel valid**. Read-write. Reset: 0. BIOS: See 2.9.9.4. 1=Communication between the DCT and DRAM phy is enabled and the DCT exits direct response mode. 0=Communication between the DCT and phy is disabled and direct response mode is enabled.

D18F2x7C_dct[0] DRAM Initialization

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

BIOS can directly control the DRAM initialization sequence using this register. To do so, BIOS sets EnDramInit to start DRAM initialization. BIOS should then complete the initialization sequence specified in the appropriate JEDEC specification. After completing the sequence, BIOS clears EnDramInit to complete DRAM initialization. BIOS should not assert LDTSTOP_L while EnDramInit is set. Setting more than one of the command bits in this register (SendControlWord, SendMrsCmd, and SendAutoRefresh) at a time results in unde-

fined behavior.

Bits	Description
31:30	Reserved.
29	**SendZQCmd: send ZQ command**. Read; Write-1-only; Cleared-by-hardware. 1=The DCT sends the ZQ calibration command to the CS or CSes specified by in D18F2xA8_dct[0][MrsCtrlWordCS]. This bit is cleared by the hardware after the command completes.
28	**AssertCke: assert CKE**. Read-write. Setting this bit causes the DCT to assert the CKE pins. This bit cannot be used to deassert the CKE pins.
27	Reserved.
26	**SendMrsCmd: send MRS command**. Read; Write-1-only; Cleared-by-hardware. 1=The DCT sends the MRS commands defined by the MrsAddress and MrsBank fields of this register to the chip selects defined in D18F2xA8_dct[0][MrsCtrlWordCS]. This bit is cleared by hardware after the command completes.
25	**SendAutoRefresh: send auto refresh command**. Read; Write-1-only; Cleared-by-hardware. 1=The DCT sends an auto refresh command. This bit is cleared by hardware after the command completes.
24	**DeassertCke: deassert CKE**. Read; Write-1-only; Cleared-by-hardware. Setting this bit causes the DCT to deassert the CKE pins. This bit cannot be used to assert the CKE pins. If the link between the DCT and the phy is not connected (D18F2x78_dct[0][PtrInitReq] has not previously been set), then this bit has no effect on the pin state until it is connected with PtrInitReq. DeassertCke register bit is cleared by hardware immediately after the register write completes. Software sets D18F2x90_dct[0][ExitSelfRef] to subsequently assert CKE.
23	**SendPDAMrsCmd: send per dram addressable MRS command**. Read; Write-1-only; Cleared-by-hardware. 1=The DCT sends the PDA MRS commands defined by the MrsAddress and MrsBank fields of this register to the chip selects defined in D18F2xA8_dct[0][MrsCtrlWordCS] with the DQ contents specified by D18F2x280_dct[0][DQPatOvrEn]. This bit is cleared by hardware after the command completes. Reserved if !Ddr4Mode.
22	Reserved.
21:18	**MrsBank[3:0]: bank address for MRS commands**. Read-write. Specifies the data driven on the DRAM bank pins for MRS commands. If Ddr3Mode, MrsBank[3] is reserved and MrsBank[2:0] corresponds to BA[2:0]. If Ddr4Mode, MrsBank[3:0] corresponds to {BG[1:0], BA[1:0]}.
17:0	**MrsAddress[17:0]: address for MRS commands**. Read-write. Specifies the data driven on the DRAM address pins for MRS commands. If Ddr4Mode, MrsAddress[16:14] are reserved.

D18F2x80_dct[0] DRAM Bank Address Mapping

IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. These fields specify DIMM configuration information. These fields are required to be programmed based on the DRAM device size and with information of the DIMM. Table 116 shows the bit numbers for each position. Table 117 shows the mapping for Ddr4Mode.

Bits	Description
31:16	Reserved.
15:12	**DimmAddrMap3: DIMM 3 address map**.
11:8	**DimmAddrMap2: DIMM 2 address map**.

7:4	**DimmAddrMap1: DIMM 1 address map**.
3:0	**DimmAddrMap0: DIMM 0 address map**.

Table 116: DDR3 DRAM Address Mapping

Bits	CS Size	Device size, width	Bank			Address																
			2	1	0		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0000b		Reserved				Row																
						Col																
0001b	256MB	512Mb, x16	15	14	13	Row	x	x	x	x	17	16	27	26	25	24	23	22	21	20	19	18
						Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
0010b	512MB	512Mb, x8	15	14	13	Row	x	x	x	17	16	28	27	26	25	24	23	22	21	20	19	18
		1Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
0011b		Reserved				Row																
						Col																
0100b		Reserved				Row																
						Col																
0101b	1GB	1Gb, x8	15	14	13	Row	x	x	17	16	29	28	27	26	25	24	23	22	21	20	19	18
		2Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
0110b		Reserved																				
0111b	2GB	2Gb, x8	15	14	13	Row	x	17	16	30	29	28	27	26	25	24	23	22	21	20	19	18
		4Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
1000b		Reserved																				
1001b		Reserved																				
1010b	4GB	4Gb, x8	15	14	13	Row	17	16	31	30	29	28	27	26	25	24	23	22	21	20	19	18
		8Gb, x16				Col	x	x	x	x	x	AP	12	11	10	9	8	7	6	5	4	3
1011b	8GB	8Gb, x8	16	15	14	Row	17	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
						Col	x	x	x	x	13	AP	12	11	10	9	8	7	6	5	4	3
1111b-1100b		Reserved																				

Table 117: DDR4 DRAM Address Mapping

Bits	CS Size	Device size, width	BG		BA		Address																		
			1	0	1	0		17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0000b		Reserved																							
0001b	1GB	2Gb, x16	x	15	14	13	Row	x	x	x	x	17	16	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
0010b	2GB	2Gb, x8	16	15	14	13	Row	x	x	x	x	17	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
0011b	2GB	4Gb, x16	x	15	14	13	Row	x	x	x	17	16	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3

Table 117: DDR4 DRAM Address Mapping (Continued)

Bits	CS Size	Device size, width	BG		BA		Address																		
			1	0	1	0		17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0100b	4GB	4Gb, x8	16	15	14	13	Row	x	x	x	17	31	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
0101b	4GB	8Gb, x16	x	15	14	13	Row	x	x	17	16	31	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
0110b	8GB	8Gb, x8	16	15	14	13	Row	x	x	17	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
0111b	8GB	16Gb x16	x	15	14	13	Row	x	17	16	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
1000b	16GB	16Gb, x8	16	15	14	13	Row	x	17	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
							Col	x	x	x	x	x	x	x	x	12	11	10	9	8	7	6	5	4	3
1111b-1001b		Reserved																							

D18F2x84_dct[0] DRAM MRS

Reset: 0000_0005h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:24	Reserved.
23	**PchgPDModeSel: precharge power down mode select**. Read-write. BIOS: Ddr3Mode. Specifies how a chip select enters and exits power down mode. This mode is enabled by D18F2x94_dct[0][PowerDownEn] and its behavior varies based on the setting of D18F2x94_dct[0][PowerDownMode]. Behavior also varies by the MR0[PPD] in D18F2x2[F4:E8]_dct[0]_mp[1:0][MxMr] in Ddr3Mode. IF (Ddr3Mode) THEN PowerDownMode / PchgPDModeSel / MR0[PPD] / Description 0b / 0b / 0b / Full channel slow exit (DLL off) 0b / 0b / 1b / Full channel fast exit (DLL on) 0b / 1b / Xb / Full channel dynamic fast exit/slow exit 1b / 0b / 0b / Reserved 1b / 0b / 1b / Partial channel fast exit (DLL on) 1b / 1b / Xb / Partial channel dynamic fast exit/slow exit ELSE PowerDownMode / PchgPDModeSel / Description 0b / 0b / Full channel fast exit 0b / 1b / Reserved 1b / 0b / Partialchannel fast exit 1b / 1b / Reserved ENDIF. See D18F2x248_dct[0]_mp[1:0][Txpdll, Txp]. In dynamic fast exit/slow exit power down mode, the DCT dynamically issues MRS command(s) to the DRAM to specify the powerdown mode; the DCT specifies fast exit mode when chip selects on one of the CKEs has recently been active; it specifies slow exit power down when chip selects on all CKEs have been idle.

22:2	Reserved.
1:0	**BurstCtrl: burst length control**. Read-write. BIOS: 01b. Specifies the number of sequential beats of DQ related to one read or write command. See MR0. Bits Description 00b 8 beats 01b Dynamic 4 or 8 beats 11b-10b Reserved

D18F2x88_dct[0] DRAM Timing Low

Reset: 3F00_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:30	Reserved.
29:24	**MemClkDis: MEMCLK disable**. Read-write. 1=Disable the MEMCLK. 0=Enable MEMCLK. All enabled clocks should be 0; all no-connect and unused clocks should be 1. Bit Pad [0] MEMCLK_H[0] [1] MEMCLK_H[1] [2] MEMCLK_H[2] [3] MEMCLK_H[3] [4] MEMCLK_H[4] [5] MEMCLK_H[5]
23:0	Reserved.

D18F2x8C_dct[0] DRAM Timing High

Bits	Description
31:19	Reserved.
18	**DisAutoRefresh: disable automatic refresh**. Read-write. Reset: 0. BIOS: 2.9.9.4. 1=Automatic refresh is disabled. 0=Automatic refresh is enabled.
17:0	Reserved.

D18F2x90_dct[0] DRAM Configuration Low

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:28	**IdleCycLimit: idle cycle limit**. Read-write. Reset: 8h. BIOS: 8h. Specifies the number of MEMCLK cycles an idle page is open before it is closed if DynPageCloseEn == 0. This field is ignored if DynPageCloseEn == 1. Bits Description 0h 8 clocks Fh-1h <IdleCycLimit>*16 clocks

27	**DisDllShutdownSR: disable DLL shutdown in self-refresh mode**. Read-write. Reset: 1. BIOS: 0. 1=Disable the power saving features of shutting down DDR phy DLLs during DRAM self refresh and memory P-states. 0=Shutdown DLLs during DRAM self refresh and allow memory P-state transitions.
26	Reserved.
25	**PendRefPaybackS3En: pending refresh payback S3 enable**. Read-write. Reset: 0. BIOS: 1. Specifies the S3 refresh payback behavior when PendRefPayback == 0. 1=Pending refreshes are paid back on S3 entry. 0=Pending refreshes are not paid back on S3 entry.
24	**StagRefEn: Stagger Refresh Enable**. Read-write. Reset: 0. BIOS: 1. 1=The DRAM controller arbitrates refreshes among chip selects based on the Tstag value. See D18F2x228_dct[0]. 0=DCT arbitrates among chip selects using the Trfc value.
23	**ForceAutoPchg: force auto precharging**. Read-write. Reset: 0. BIOS: See 2.9.9.4. 1=Force auto-precharge cycles with every read or write command.
22:21	**IdleCycLowLimit: idle cycle low limit**. Read-write. Reset: 0. Specifies the number of MEMCLK cycles a page is allowed to be open before it may be closed by the dynamic page close logic. This field is ignored if D18F2x90_dct[0][DynPageCloseEn] == 0. Bits Description 00b 16 clocks 01b 32 clocks 10b 64 clocks 11b 96 clocks
20	**DynPageCloseEn: dynamic page close enable**. Read-write. Reset: 0. See 2.9.9.4. 1=The DRAM controller dynamically determines when to close open pages based on the history of that particular page and D18F2x90_dct[0][IdleCycLowLimit]. 0=Any open pages not auto-precharged by the DRAM controller are automatically closed after IdleCycLimit clocks of inactivity.
19	Reserved.
18	**PendRefPayback: pending refresh payback**. Read-write. Reset: 0. BIOS: 0. 1=The DRAM controller executes all pending refresh commands before entering the self refresh state. 0=The controller enters the self refresh state regardless of the number of pending refreshes; applies to any self refresh entry if PendRefPaybackS3En == 0, else any non-S3 self refresh entry.
17	**EnterSelfRef: enter self refresh command**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. 1=The DRAM controller places the DRAMs into self refresh mode. The DRAM interface is tristated 1 MEMCLK after the self refresh command is issued to the DRAMs. Once entered, the DRAM interface must remain in self refresh mode for a minimum of 5 MEMCLKs. This bit is read as a 1 while the enter-self-refresh command is executing; it is read as 0 at all other times.
16	**UnbuffDimm: unbuffered DIMM**. Read-write or Read-only, depending on the product. Reset: Product-specific. 1=The DRAM controller is connected to unbuffered DIMMs. 0=Reserved.
15:8	Reserved.
7	**DataScrambleEn: data scramble enable**. Read-write; Same-for-all. BIOS:1. Reset: 0. 1=Data scrambling enabled. Data stored in the DRAM will be scrambled. 0=Data scrambling disabled. This register must have the same value for all DCTs. Data scrambling must be disabled prior to using D18F2x250_dct[0] to generate patterns. BIOS should set this bit prior to any memory write transactions to DRAM after training has completed; see D18F2x110[MemClrInit].
6:2	Reserved.

1	**ExitSelfRef: exit self refresh (after suspend to RAM or for DRAM training) command**. Read, Write-1-only; Cleared-by-hardware. Reset: 0. Writing a 1 to this bit causes the DRAM controller to bring the DRAMs out of self refresh mode. It also causes the DRAM controller to issue ZQCL and MRS commands per D18F2x2E0_dct[0][MxMrsEn] for the current memory P-state if D18F2x78_dct[0][ChanVal] == 0. Software should set D18F2x78_dct[0][PtrInitReq] prior to this bit. This command should be executed by BIOS when returning from the suspend to RAM state, after the DRAM controller configuration registers are properly initialized, or when self refresh is used during DRAM training. This bit is read as 1 while the exit-self-refresh command is executing; and is read as 0 at all other times. This bit should not be set if the DCT is disabled.
0	Reserved.

D18F2x94_dct[0] DRAM Configuration High

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31	**DphyMemPsSelEn**. Read-write. Reset: 1. BIOS: 0. 1=The DCT uses D18F1x10C[MemPsSel] to configure DctOffset[24] to the phy while the value that software writes to DctOffset[24] is ignored. 0=Software determines DctOffset[24]. BIOS must clear this bit for proper operation.
30	Reserved.
29	**BankGroupSwap: bank group swap**. Read-write. Reset: 0. BIOS: 2.9.9.4. 1=Swap normalized address bits[7:6] with the bank group bits. Reserved if !Ddr4Mode.
28:24	**DcqBypassMax: DRAM controller queue bypass maximum**. Read-write. Reset: 00h. BIOS: 2.9.9.4. The DRAM controller arbiter normally allows transactions to pass other transactions in order to optimize DRAM bandwidth. This field specifies the maximum number of times that the oldest memory-access request in the DRAM controller queue may be bypassed before the arbiter decision is overridden and the oldest memory-access request is serviced instead. Bits — Description 00h — No bypass; the oldest request is never bypassed. 1Fh-01h — The oldest request may be bypassed no more than <DcqBypassMax> time.
23	**On3RdCasStallMode: onion read CAS stall mode**. Read-write. Reset: 0. BIOS: ~(D18F2xA8_dct[0][FastSelfRefEntryDis] & ~D18F5xA0[DisCasBasedOpbdMgmt]); 0=Stall all commands. 1=Stall only Onion3 read commands.
22	**BankSwizzleMode: bank swizzle mode**. Read-write. Reset: 0. BIOS: 2.9.9.4. 1=Remaps the DRAM device bank address bits as a function of normalized physical address bits. Each of the bank address bits, as specified in D18F2x80_dct[0], are remapped as follows: • Define X as a bank address bit (e.g., X = 15 if the bank bit is specified to be address bit[15]). • Define S(n) as the state of address bit[n] (0 or 1) and B as the remapped bank address bit. Then, Ddr3Mode: B= S(X) ^ S(X + 3) ^ S(X + 6); for an 8-bank DRAM. Ddr4Mode: B= S(X) ^ S(X + 3) ^ S(X + 6); for an 8-bank DRAM. B= S(X) ^ S(X + 4) ^ S(X + 8); for a 16-bank DRAM. For example, encoding 02h of Table 116 would be remapped from Bank[2:0] = {A15, A14, A13} to the following: Bank[2:0] = {A15^A18^A21, A14^A17^A20, A13^A16^A19}.
21	Reserved.

20	**SlowAccessMode: slow access mode (a.k.a. 2T mode)**. Read-write. Reset: 0. 1=One additional MEMCLK of setup time is provided on all DRAM address and control signals (not including CS, CKE, and ODT; i.e., these signals are driven for two MEMCLK cycles rather than one). 0=DRAM address and control signals are driven for one MEMCLK cycle. 2T mode may be needed in order to meet electrical requirements of certain DIMM speed and loading configurations. If memory P-states are enabled then BIOS must set this bit if 2T timing is recommended for either memory P-state.
19:17	Reserved.
16	**PowerDownMode: power down mode**. Read-write. Reset: 0. BIOS: 1. Specifies how a chip select or group of chip selects enters power down mode when enabled by D18F2x94_dct[0][PowerDownEn]. A chip select enters power down mode when the DCT deasserts the CKE pin. The DCT behavior varies based on the setting of D18F2x84_dct[0][PchgPDModeSel]. See also Table 115 [DIMM, Chip Select, and Register Mapping]. 0=Channel CKE control mode; the DRAM channel is placed in power down mode when all chip selects associated with the channel are idle; CKE pins for the channel operate in lock step in terms of placing the channel in power down mode. 1=Chip select CKE control mode; the chip select group controlled by a CKE pinis placed in power down mode when all chip selects associated with the pin are idle.
15	**PowerDownEn: power down mode enable**. Read-write. Reset: 0. BIOS: 1. 1=Power down mode is enabled. Only precharge power down mode is supported, not active power down mode. See PowerDownMode, D18F2x84_dct[0][PchgPDModeSel], D18F2xA8_dct[0][PrtlChPDEnhEn, AggrPDEn, PDPhyPSDis], and D18F2x248_dct[0]_mp[1:0][PchgPDEnDelay].
14	**DisDramInterface: disable the DRAM interface**. Read-write. Reset: 0. 1=The DRAM controller is disabled and the DRAM interface is placed into a low power state. This bit must be set if there are no DIMMs connected to the DCT.
13	Reserved.
12:10	**ZqcsInterval: ZQ calibration short interval**. Read-write. Reset: 000b. This field specifies the programmable interval for the controller to send out the DRAM ZQ calibration short command. Bits Description 0h ZQ calibration short command is disabled 1h 64 ms 2h 128 ms 3h 256 ms 7h-4h Reserved.
9:8	Reserved.
7	**MemClkFreqVal: memory clock frequency valid**. Read-write. Reset: 0. System BIOS should set this bit after setting up D18F2x94_dct[0][MemClkFreq] to the proper value. This indicates to the DRAM controller that it may start driving internal channel clocks corresponding to MEMCLK to the proper frequency. This bit should not be set if the DCT is disabled. BIOS must change each DCT's operating frequency in order.
6:5	Reserved.
4:0	**MemClkFreq: memory clock frequency**. Read-write. Reset: 000b. Specifies the frequency and rate of the DRAM interface (MEMCLK). See: Table 118 [Valid Values for Memory Clock Frequency Value Definition]. The rate is twice the frequency. See D18F5x84[DdrMaxRate] and D18F5x84[DdrMaxRateEnf]. See MemClkFreqVal.

Table 118: Valid Values for Memory Clock Frequency Value Definition

Bits	Description
01h-00h	Reserved
02h	200 MHz. (400 MT/s)
03h	Reserved
04h	333 MHz. (667 MT/s)
05h	Reserved
06h	400 MHz. (800 MT/s)
09h-07h	Reserved
0Ah	533 MHz. (1066 MT/s)
0Dh-0Bh	Reserved
0Eh	667 MHz. (1333 MT/s)
11h-0Fh	Reserved
12h	800 MHz. (1600 MT/s)
15h-13h	Reserved
16h	933 MHz. (1866 MT/s)
19h-17h	Reserved
1Ah	1066 MHz. (2133 MT/s)
1Eh-1Bh	Reserved
1Fh	1200 MHz. (2400 MT/s)

D18F2x98_dct[0] DRAM Controller Additional Data Offset

Reset: 8000_0000h. See 2.9.3 [DCT Configuration Registers].

Each DCT includes an array of registers that are used primarily to control DRAM-interface electrical parameters. Access to these registers is accomplished as follows:

Reads (without auto-increment):
1. Write the register number to D18F2x98_dct[0][DctOffset] with D18F2x98_dct[1:0][DctAccessWrite, DctOffsetAutoIncEn] = {0,0}.
2. Read the register contents from D18F2x9C_dct[0].

Writes (without auto-increment):
1. Write all 32 bits of register data to D18F2x9C_dct[0] (individual byte writes are not supported).
2. Write the register number to D18F2x98_dct[0][DctOffset] with D18F2x98_dct[0][DctAccessWrite, DctOffsetAutoIncEn] = {1,0}.
 - The data will be delivered to the PHY similar to a posted memory-write, and the write will complete with-out any further action. However, to ensure that the contents of the array register write have been deliv-ered to the PHY, software issues a subsequent configuration register read or write to any register in the northbridge. For example, reading D18F2x98_dct[0] will accomplish this.

Reads (with auto-increment): Not supported.

Writes (with auto-increment):
1. Write the first register number to D18F2x98_dct[0][DctOffset] with D18F2x98_dct[0][DctAccessWrite, DctOffsetAutoIncEn] = {1,1}.
2. Write all 32 bits of register data to D18F2x9C_dct[0].
3. Repeat step 2 as needed for each additional array register write.
4. Program D18F2x98_dct[0][DctOffsetAutoIncEn] = 0.

Bits	Description
31	Reserved.
30	**DctAccessWrite: DRAM controller read/write select**. Read-write. 0=Specifies a read access. 1=Specifies a write access.
29	**DctOffsetAutoIncEn: DCT offset auto-increment enable**. Read-write. 1=Specifies that subsequent accesses causes the DCT to increment the DctOffset field by one after the access. When in this mode, the DCT generates accesses to the PHY registers with sequential accesses only to the data port (D18F2x9C_dct[0]). 0=DctOffset is not incremented after each access.
28:0	**DctOffset: DRAM controller offset**. Read-write.

D18F2x9C_dct[0] DRAM Controller Additional Data Port

See D18F2x98_dct[0] for register access information. See 2.9.3 [DCT Configuration Registers]. Address: D18F2x98_dct[0][DctOffset].

Bits	Description
31:0	**Data**.

D18F2x9C_x00[F,3:0]0_0009_dct[0] High Addr Mode

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 119: Index Mapping for D18F2x9C_x00[F,3:0]0_0009_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:1	Reserved.
0	**HiAddrMode**. Read-write. Reserved if !Ddr3Mode. 1=CS_L[7:6] function with address timing. 0=CS_L[7:6] function with chipselect timing.

D18F2x9C_x0000_000E_dct[0] Global Control Slave

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Bits	Description
31:1	Reserved.
0	**D4_Mode.** Read-write. 1=DDR4 mode. 0=DDR3 mode.

D18F2x9C_x00[F,3:0]0_0013_dct[0] NB Pstate

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 120: Index Mapping for D18F2x9C_x00[F,3:0]0_0013_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:2	Reserved.
1:0	**NB_Pstate.** Read-write, Updated-by-hardware. Bits Description 11b-00b <NB_Pstate>

D18F2x9C_x0[3,1:0][F,3:0]0_0014_dct[0] Dll Lock Maintenance

Cold reset: 0000_0037h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 121: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_0014_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:8	Reserved.
7:4	**DllPumpCount**. Read-write. BIOS:0. Specifies the number of DLL relocks required to keep the receive DLLs locked for the period where there is no traffic.
3:0	**MaxDurDllNoLock**. Read-write. BIOS: 0. Specifies the number of UI the DLL can wait between locking events. If MaxDurDllNoLock != 0 (standby is enabled), D18F2x9C_x00[F,3:0]0_0078_dct[0][DllResetRelock] must be set to 1 prior to writing this register and then DllResetRelock must be cleared after updating MaxDurDllNoLock. Bits — Description 0h — DLL power saving(standby) disabled. 9h-1h — Relock 2^(<MaxDurDllNoLock>+1) clocks Fh-Ah — Reserved

D18F2x9C_x00F0_0015_dct[0] Vref Byte

Cold reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3:0	**VrefFilt: Vref filter.** Read-write. BIOS: 0. This field adjusts noise coupling on to VrefOut and adjusts the input resistance.

D18F2x9C_x00[F,3:0]0_001A_dct[0] ByteDbgCtrl

Cold reset: 0000_0000h.

Table 122: Index Mapping for D18F2x9C_x00[F,3:0]0_001A_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:16	Reserved.
15:8	**DbDbgSelByte1**. Read-write.
7:0	**DbDbgSelByte0**. Read-write.

D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]028_dct[0] ABYTE RdPtrOffset

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 123: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]028_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[2:0]	Timing Group
D18F2x98_dct[0][15:12]	Fh	Timing Group [2:0]
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:6	Reserved.
5:0	**TxRdPtrOffset: Tx Read Pointer Offset**. Read-write. BIOS: See 2.9.9.2.6. The amount of time UI that is added to the read pointer of the Tx command-FIFO for reading out commands.

D18F2x9C_x00[F,3:0]0_[F,2:0][8,3:0]2E_dct[0] ABYTE RdPtrInitVal

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 124: Index Mapping for D18F2x9C_x00[F,3:0]0_[F,2:0][8,3:0]2E_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][11:8]	[3:0]	NbPstate
D18F2x98_dct[0][11:8]	8h	NbPstate PMU
D18F2x98_dct[0][15:12]	[2:0]	Timing Group
D18F2x98_dct[0][15:12]	Fh	Timing Group [2:0]
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:6	Reserved.
5:0	**RdPtrInitVal: Read Pointer Initial Value**. Read-write. BIOS: See 2.9.9.2.6. Specifies RdPtr initial value for the transmit command-FIFOs.

D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] ABYTE Tx Impedance

Cold reset: 0000_0FFFh. BIOS: See 2.9.9.2.4. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 125: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]

Table 125: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0] (Continued)

D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:12	Reserved.
11:6	**DrvStrenP: PMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.4. Specifies the pull-down output driver impedance. See DrvStrenN for field description.
5:0	**DrvStrenN: NMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.4. Specifies the pull-up output driver impedance. Bits Description 00h disabled 01h 120 ohms 03h 60 ohms 07h 40 ohms 0Fh 30 ohms 1Fh 24 ohms 3Fh 20 ohms

D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0] ABYTE DqDqsRcvCntrl1

Cold reset: 0000_0100h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 126: Index Mapping for D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:13	Reserved.
12:10	**AVoltageLevel**. Read-write. Bits Description 0h Reserved 1h 1.25V 2h 1.35V 3h 1.5V 7h-4h Reserved
9	**BiasBypassEn**. Read-write.
8	**PowerDownRcvr**. Read-write. 1=Receiver Power down enabled. 0=Receiver Power down disabled.

7:5	**MajorMode: Major mode**. Read-write. BIOS: See 2.9.9.2. Specifies operating mode of the PHY logic. Bits Description 000b DDR3 010b-001b Reserved 011b DDR4 100b DDR3 low-power/low-speed 110b-101b Reserved 111b DDR4 low-power/low-speed
4:0	Reserved.

D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0] TxControlDq

Cold reset: 0000_1300h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 127: Index Mapping for D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:15	Reserved.
14	**BurstStartEqVal**. Read-write.
13	**EQMode**. Read-write.
12	**EQEnable**. Read-write.
11	**BurstStartEqEn**. Read-write.
10	**EnContEQ**. Read-write.
9	**TxCalBaseN**. Read-write. 1=Disable N Tx driver (Hi-Z). 0=Enable N Tx driver.
8	**TxCalBaseP**. Read-write. 1=Disable P Tx driver (Hi-Z). 0=Enable P Tx driver.
7:0	Reserved.

D18F2x9C_x00[F,3:0]0_[F,B:0]050_dct[0] TxControlDq2

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 128: Index Mapping for D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet

Table 128: Index Mapping for D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0] (Continued)

D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:16	Reserved.
15	**DrvPwrGateEn**. Read-write.
14:0	Reserved.

D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0] ABYTE Tx Slew Rate

Cold reset: 0000_00FFh. BIOS: See 2.9.9.2.4. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 129: Index Mapping for D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:8	Reserved.
7:4	**TxPreN: NMOS predriver code**. Read-write. BIOS: See 2.9.9.2.7. Specifies the falling edge slew rate of the transmit pad. 0000b=Slowest slew rate. 1111b=Fastest slew rate.
3:0	**TxPreP: PMOS predriver code**. Read-write. BIOS: See 2.9.9.2.7. Specifies the rising edge slew rate of the transmit pad. 0000b=Slowest slew rate. 1111b=Fastest slew rate.

D18F2x9C_x0[3,1:0][F,3:0]0_006D_dct[0] CkPhaseCtl

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 130: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_006D_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:14	Reserved.
13	**RDIMMForceCKPhyInit**. Read-write. 1=All CK_t/CK_c lanes forced low. 0=All CK_t/CK_c lanes tristate.
12:0	Reserved.

D18F2x9C_x00[F,3:0]0_0077_dct[0] DllPowerdown

Cold reset: 0000_0000h. BIOS: See 2.9.9.9. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 131: Index Mapping for D18F2x9C_x00[F,3:0]0_0077_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:13	Reserved.
12	**DllAltPiClkInEn**. Read-write.
11	**DllAltPiClkOutEn**. Read-write.
10:7	Reserved.
6	**DllPowerDownTx**. Read-write.
5	**DllPowerDownXCLK**. Read-write.
4	Reserved.
3:1	**DllPowerDownPI**. Read-write.
0	**DllPowerDown**. Read-write.

D18F2x9C_x00[F,3:0]0_0078_dct[0] DllControl

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 132: Index Mapping for D18F2x9C_x00[F,3:0]0_0078_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]

Bits	Description
31:8	Reserved.
7	**DllResetRelock**. Read-write, Updated-by-hardware. 1=Reset the DLL. 0=Relock the DLL. This bit must be set for at least 20 ns and then cleared anytime a forced relock of the DLL is required.
6:0	Reserved.

D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]081_dct[0] ABYTE TxDly

Cold reset: 0000_0000h. BIOS: See 2.9.9.2.10. This register controls the timing of the address, command, chip select, ODT and clock enable pins with respect to memory clock. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 133: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]081_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[2:0]	Timing Group
D18F2x98_dct[0][15:12]	Fh	Timing Group [2:0]

Table 133: Index Mapping for D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]081_dct[0] (Continued)

D18F2x98_dct[0][23:20]	[3:0]	ABYTE chiplet
D18F2x98_dct[0][23:20]	Fh	ABYTE chiplet [3:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:5	Reserved.
4:0	**TxDly: Tx fine delay**. Read-write. Specifies the time that the address/command signals are delayed from the default setup time, in increments of 1/32 UI.

D18F2x9C_x00[F,8:0]1_0000_dct[0] VariousChicken

Cold reset: 0000_0000h. BIOS: See 2.9.9.9. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 134: Index Mapping for D18F2x9C_x00[F,8:0]1_0000_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:3	Reserved.
2	**DByteEnable: data byte enable**. Read-write. Controls whether this DBYTE is enabled. If this DBYTE is not enabled, it receives no clocks and remains in reset. 1=Disable this DBYTE. 0=Enable this DBYTE.
1:0	Reserved.

D18F2x9C_x00[F,8:0]1_000E_dct[0] Global Control Slave

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 135: Index Mapping for D18F2x9C_x00[F,8:0]1_000E_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:1	Reserved.
0	**D4_Mode**. Read-write. 1=DDR4 mode. 0=DDR3 mode.

D18F2x9C_x0[3,1:0][F,8:0]1_0010_dct[0] DByteCtrl1

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 136: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0010_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:5	Reserved.
4	Reserved. Read-write.
3:0	**X4Dimm**. Read-write. Specifies this DIMM slot is x4. If D18F2xA8_dct[0][PerRankTimingEn] == 1 then the function is CS, otherwise the function is DIMM. Bit Description [0] DIMM/CS 0 [1] DIMM/CS 1 [2] DIMM/CS 2 [3] DIMM/CS 3

D18F2x9C_x00[F,8:0]1_0013_dct[0] NB Pstate

Cold reset: 0000_0000h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 137: Index Mapping for D18F2x9C_x00[F,8:0]1_0013_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:2	Reserved.
1:0	**NB_Pstate.** Read-write, Updated-by-hardware. Bits Description 11b-00b <NB_Pstate>

D18F2x9C_x0[3,1:0][F,8:0]1_0014_dct[0] Dll Lock Maintenance

Cold reset: 0000_0037h. See 2.9.4.1 [DDR Chip to Pad Mapping].

Table 138: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0014_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:8	Reserved.
7:4	**DllPumpCount**. Read-write. BIOS: 0. Specifies the number of DLL relocks required to keep the receive DLLs locked for the period where there is no traffic.
3:0	**MaxDurDllNoLock**. Read-write. BIOS: 0. Specifies the number of UI the DLL can wait between locking events. If MaxDurDllNoLock != 0 (standby is enabled), D18F2x9C_x00[F,8:0]1_0[F,2:0]78_dct[0][DllResetRelock] must be set to 1 prior to writing this register and then DllResetRelock must be cleared after updating MaxDurDllNoLock. Bits — Description 0h — DLL power saving(standby) disabled. 9h-1h — Relock 2^(<MaxDurDllNoLock>+1) clocks Fh-Ah — Reserved

D18F2x9C_x00[F,8:0]1_0015_dct[0] Vref Byte

Cold reset: 0000_0000h.

Table 139: Index Mapping for D18F2x9C_x00[F,8:0]1_0015_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:4	Reserved.
3:0	**VrefFilt: Vref filter.** Read-write. BIOS: 0. This field adjusts noise coupling on to VrefOut and adjusts the input resistance.

D18F2x9C_x00[F,8:0]1_0016_dct[0] Proc Odt Timing

Cold reset: 0000_1244h. See 2.9.4.1 for chiplet to pin mapping.

Table 140: Index Mapping for D18F2x9C_x00[F,8:0]1_0016_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:16	Reserved.
15	**ProcOdtOn: Proc ODT on**. Read-write. BIOS: 0. 1=If Ddr3Mode then PHY enables processor ODT when Tx not driving. If Ddr4Mode drive a high level through 60 ohm ODT.
14	**ProcOdtOff: Proc ODT off**. Read-write. BIOS: 0. 1=PHY never enables ODT unless ProcOdtOn == 1.
13:11	**POdtStartDelayDqs: Proc ODT start delay DQS**. Read-write. BIOS: 2h. Controls the ODT turn on delay for DQS during reads. Bits: Delay 100b-000b: <2*POdtStartDelayDqs> UI 110b-101b: Reserved. 111b: Uses RxTraffic to turn-on at the earliest possible time and stays enabled for duration of DLL operation.
10:8	**POdtStartDelayDq: Proc ODT start delay DQ**. Read-write. BIOS: 2h. Controls the ODT turn on delay for DQ during reads. Bits: Delay 100b-000b: <2*POdtStartDelayDq> UI 110b-101b: Reserved. 111b: Uses RxTraffic to turn-on at the earliest possible time and stays enabled for duration of DLL operation.
7:4	**POdtWidthDqs: Proc ODT width DQS**. Read-write. BIOS: 4h. Controls the duration of ODT for DQS during reads. Bits: Duration 0111b-0000b: <8+(2*POdtWidthDqs)> UI 1111b-1000b: Reserved.
3:0	**POdtWidthDq: Proc ODT width DQ**. Read-write. BIOS: 4h. Controls the duration of ODT for DQ during reads. Bits: Duration 0111b-0000b: <8+(2*POdtWidthDq)> UI 1111b-1000b: Reserved.

D18F2x9C_x00[F,8:0]1_001A_dct[0] ByteDbgCtrl

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 141: Index Mapping for D18F2x9C_x00[F,8:0]1_001A_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:16	Reserved.
15:8	**DbDbgSelByte1**. Read-write.
7:0	**DbDbgSelByte0**. Read-write.

D18F2x9C_x00[F,8:0]1_001C_dct[0] DynPwrDnUp

Cold reset: 0000_0001h. See 2.9.4.1 for chiplet to pin mapping.

Table 142: Index Mapping for D18F2x9C_x00[F,8:0]1_001C_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:1	Reserved.
0	**DynPowerDown: Dynamic Power Down**. Read-write, Updated-by-hardware. 1=Analog circuitryis turned off. 0=Normal operation.

D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0] DATA RdPtrOffset

Cold reset: 0000_0004h. See 2.9.4.1 for chiplet to pin mapping.

Table 143: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:11	Reserved.

10:6	**TxRdPtrOffset: Tx Read Pointer Offset**. Read-write. BIOS: See 2.9.9.2.6. The amount of time (specified in units of 2UI) that is added to the read pointer of the TX FIFO for reading out TX data from the TX data-FIFO.
5:0	**RxRdPtrOffset: Rx Read Pointer Offset**. Read-write. BIOS: See 2.9.9.2.6. The amount of time (specified in units of 2UI) that is added to the read pointer of the TX FIFO for reading out received data from the RX data-FIFO.

D18F2x9C_x0[3,1:0][F,8:0]1_0029_dct[0] EarlyTrafficOffset

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 144: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_0029_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:11	Reserved.
10:6	**TxTrafficOffset**. Read-write. Specifies the amount of time that TxEarlyTraffic is asserted prior to TxEn, unit=2*UI.
5:0	**RxTrafficOffset**. Read-write. Specifies the amount of time that RxEarlyTraffic is asserted prior to RxEn, unit=2*UI.

D18F2x9C_x0[3,1:0][F,8:0]1_002A_dct[0] RxStaggerCnt

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 145: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_002A_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:12	Reserved.

11:6	**RxStggrPost: Rx stagger postamble**. Read-write. Specifies the duration RX Dlls remain locked after read request (RxEn), unit=4*UI.
5:0	**RxStggrAnte: Rx stagger preamble**. Read-write. Specifies the duration RX Dlls remain locked after RxEarlyTraffic, unit=4*UI.

D18F2x9C_x0[3,1:0][F,8:0]1_002B_dct[0] Tx Dll Standby Stagger Config

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 146: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_002B_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:10	Reserved.
9:5	**TxStggrPost: Tx stagger postamble**. Read-write. Specifies the duration TX Dlls remain locked after write request (TxEn), unit=4*UI.
4:0	**TxStggrAnte: Tx stagger preamble**. Read-write. Specifies the duration TX Dlls remain locked after TxEarlyTraffic, unit=4*UI.

D18F2x9C_x0[3,1:0][F,8:0]1_002C_dct[0] Rx Pad Traffic Early Offset

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 147: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_002C_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:6	Reserved.
5:0	**RxPadTrafficOffset: Rx pad traffic offset**. Read-write. Specifies the duration RxPadEarlyTraffic is asserted prior to RxEn, unit=2*UI.

D18F2x9C_x00[F,8:0]1_0[8,3:0]2E_dct[0] DATA RdPtrInitVal

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pin mapping.

Table 148: Index Mapping for D18F2x9C_x00[F,8:0]1_0[8,3:0]2E_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][11:8]	[3:0]	NbPstate
D18F2x98_dct[0][11:8]	8h	NbPstate PMU
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:7	Reserved.
6:2	**RdPtrInitVal[6:2]: Rd pointer initial value[6:2]**. Read-write. BIOS: See 2.9.9.2.6. Specifies RdPtr initial value for the transmit FIFOs. Each RdPtrInitVal[6:0] is in units of UI. Software may write entire RdPtrInitVal[6:0] field.
1:0	**RdPtrInitVal[1:0]: Rd pointer initial value[1:0]**. RAZ. BIOS: See 2.9.9.2.6. Specifies RdPtr initial value for the transmit FIFOs. Each RdPtrInitVal[6:0] is in units of UI. Software may write entire RdPtrInitVal[6:0] field.

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0] TxImpedanceDq

Cold reset: 0000_1FFFh. See 2.9.4.1 for chiplet to pin mapping.

Table 149: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][25:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][25:24]	3h	Memory Pstate [1:0]

Bits	Description
31:14	Reserved.

13:7	**DrvStrenN: NMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.5. Specifies the pull-up output driver impedance. See DrvStrenP.
6:0	**DrvStrenP: PMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.5. Specifies the pull-down output driver impedance. Bits Description — Bits Description 00h Disabled — 33h-32h Reserved 01h 480 ohms — 34h 48 ohms 03h-02h Reserved — 35h 43.6 ohms 04h 240 ohms — 69-36h Reserved 05h 160 ohms — 70h 40 ohms 0Bh-06h Reserved — 71h 36.9 ohms 0Ch 120 ohms — 73-72h Reserved 0Dh 96 ohms — 74h 34.3 ohms 0Eh Reserved — 75h 32 ohms 0Fh 80 ohms — 7Bh-76h Reserved 14h-10h Reserved — 7Ch 30 ohms 15h 68 ohms — 7Dh 28.2 ohms 29h-16h Reserved — 7Eh Reserved 30h 60 ohms — 7Fh 26.7 ohms 31h 53.3 ohms

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]043_dct[0] DqDqsRcvCntrl2

Cold reset: 0000_0000h. See 2.9.4.1.

Table 150: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]043_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:5	Reserved.
4	**ExtVrefRange**. Read-write. BIOS: Ddr4Mode. 1=Extend the Vref range. 0=Normal Vref range.
3	**SelExternalVref**. Read-write. 1=Select external Vref. 0=Select internal Vref.
2:0	**MajorMode: Rx Major mode**. Read-write. BIOS: See 2.9.9.9. Bits Description 000b DDR3 normal 001b Reserved 011b DDR4 100b DDR3 low power 111b-101b Reserved

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]044_dct[0] VrefHspeed

Cold reset: 0000_0000h. See 2.9.4.1.

Table 151: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]044_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:7	Reserved.
6:0	**VrefHspeed**. Read-write.

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]045_dct[0] VrefLpower

Cold reset: 0000_0000h. See 2.9.4.1.

Table 152: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]045_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:7	Reserved.
6:0	**VrefLpower**. Read-write.

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]046_dct[0] TxStrenHi

Cold reset: 0000_0000h. See 2.9.4.1.

Table 153: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]046_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:14	Reserved.
13:7	**EQStrenHiN: Equalization HI NMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.10. Specifies the pulldown output driver impedance during Hi de-emphasis.
6:0	**EQStrenHiP: Equalization HI PMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.10. Specifies the pullup output driver impedance during Hi de-emphasis. EQStrenHiP[6:4] is the number of additional 120 Ohm pull ups {i.e., POPCNT(EQStrenHiP[6:4]) equals 0, 1, 2, 3 pullups} in parallel with those specified by ODTStrenP.

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]047_dct[0] TxStrenLo

Cold reset: 0000_0000h. See 2.9.4.1.

Table 154: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]047_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad[B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:14	Reserved.

13:7	**EQStrenLoN: Equalization LO NMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.10. Specifies the pulldown output driver impedance during Lo de-emphasis.
6:0	**EQStrenLoP: Equalization LO PMOS driver output impedance**. Read-write. BIOS: See 2.9.9.2.10. Specifies the pullup output driver impedance during Lo de-emphasis.

D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0] DqDqsRcvCntrl1

Cold reset: 0000_0000h. See 2.9.4.1.

Table 155: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad [B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:13	Reserved.
12:10	**VoltageLevel**. Read-write. Specifies voltage level to the DQS receiver. Unused in DQ receiver. Bits Description 0h Reserved 1h 1.25V 2h 1.35V 3h 1.5V 7h-4h Reserved
9	**BiasBypassEn**. Read-write.
8	**PowerDownRcvr**. Read-write. 1=Receiver Power down enabled. 0=Receiver Power down disabled.
7:5	Reserved.
4:0	**DlyCtrl: Fine delay control**. Read-write. Controls the delay setting of the fine delay block. Bits Description 0h Minimum delay Fh Maximum delay 1Fh-10h Reserved

D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] DATA Rx Impedance

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pad and pad to pin mapping.

Table 156: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad [B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet

Table 156: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0] (Continued)

D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:8	Reserved.
7:4	**ODTStrenN: ODT strength NMOS**. Read-write. BIOS: See 2.9.9.2.5. Specifies the ODT impedance when in thevinin termination mode. See ODTStrenP. This field should be set to 0000b in DDR4 mode.
3:0	**ODTStrenP: ODT strength PMOS**. Read-write. BIOS: See 2.9.9.2.5. Specifies the ODT impedance when in thevinin termination mode. Note legacy ProcOdt values are given in Thevenin. Bits — Description 0h — Disabled 1h — 480 ohms 3h-2h — Reserved 4h — 240 ohms 5h — 160 ohms Bh-6h — Reserved Ch — 120 ohms Dh — 96 ohms Eh — Reserved Fh — 80 ohms

D18F2x9C_x00[F,8:0]1_[F,B:0]04E_dct[0] TxControlDq

Cold reset: 0000_1300h. See 2.9.4.1.

Table 157: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]04E_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad [B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:15	Reserved.
14	**BurstStartEqVal**. Read-write.
13	**EQMode**. Read-write.
12	**EQEnable**. Read-write.
11	**BurstStartEqEn**. Read-write.

10	**EnContEQ**. Read-write.
9	**TxCalBaseN**. Read-write. 1=Disable N Tx driver (Hi-Z). 0=Enable N Tx driver.
8	**TxCalBaseP**. Read-write. 1=Disable P Tx driver (Hi-Z). 0=Enable P Tx driver.
7:0	Reserved.

D18F2x9C_x00[F,8:0]1_[F,B:0]050_dct[0] TxControlDq2

Cold reset: 0000_0000h. See 2.9.4.

Table 158: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]050_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad [B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:16	Reserved.
15	**DrvPwrGateEn**. Read-write.
14:0	Reserved.

D18F2x9C_x00[F,8:0]1_[F,B:0]051_dct[0] DqDqsRcvCntrl3

Cold reset: 0000_0012h. See 2.9.4.

Table 159: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]051_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad [B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:7	Reserved.
6	**RxPadStandbyEn: Rx pad standby enable**. Read-write. BIOS: 1. 1=Enables standby power savings for the receiver.
5	Reserved.
4:0	**GainCurrAdj.** Read-write.

D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0] DATA Tx Slew Rate

Cold reset: 0000_00FFh. See 2.9.4.1.

Table 160: Index Mapping for D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][15:12]	[B:0]	pad
D18F2x98_dct[0][15:12]	Fh	pad [B:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:9	Reserved.
8	**D4Mode**. Read-write. BIOS: 0. 1=Enable DDR4 test mode. 0=Disable DDR4 test mode.
7:4	**TxPreN: NMOS predriver code**. Read-write. BIOS: See 2.9.9.2.7. Specifies the falling edge slew rate of the transmit pad. 0000b=Slowest slew rate. 1111b=Fastest slew rate.
3:0	**TxPreP: PMOS predriver code**. Read-write. BIOS: See 2.9.9.2.7. Specifies the rising edge slew rate of the transmit pad. 0000b=Slowest slew rate. 1111b=Fastest slew rate.

D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0] DllPowerdown

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pad mapping and see 2.9.4 for pad to pin mapping.

Table 161: Index Mapping for D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][11:8]	[2:0]	DLL
D18F2x98_dct[0][11:8]	Fh	DLL [2:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:13	Reserved.
12	**DllAltPiClkInEn**. Read-write.
11	**DllAltPiClkOutEn**. Read-write.
10:6	Reserved.
5	**DllPowerDownXCLK**. Read-write.
4	Reserved.
3:1	**DllPowerDownPI**. Read-write.
0	**DllPowerDown**. Read-write.

D18F2x9C_x00[F,8:0]1_0[F,2:0]78_dct[0] DllControl

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pad mapping and see 2.9.4 for pad to pin mapping.

Table 162: Index Mapping for D18F2x9C_x00[F,8:0]1_0[F,2:0]78_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][11:8]	[2:0]	DLL
D18F2x98_dct[0][11:8]	Fh	DLL [2:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]

Bits	Description
31:8	Reserved.
7	**DllResetRelock: Dll reset relock**. Read-write. 1=Reset the DLL. 0=Relock the DLL. This bit must be set for 20 ns and then cleared anytime a forced relock of the DLL is required.
6:0	Reserved.

D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0] Rx Delay

Cold reset: 0000_0040h. See 2.9.4.1 for chiplet to pad mapping and see 2.9.4 for pad to pin mapping.

Table 163: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][11:8]	[3:0]	DIMM
D18F2x98_dct[0][11:8]	Fh	DIMM[3:0]
D18F2x98_dct[0][15:12]	[3:0]	Timing Group
D18F2x98_dct[0][15:12]	Fh	Timing Group [3:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:10	Reserved.
9:0	**RxDly: Rx delay**. Read-write. BIOS: See 2.9.9.2.10. Specifies the time that the DATA (includes both DQ and DQS/WCK) chiplet signals are delayed from the default setup time, in increments of 1/32 UI. When applied to a CSR for a data strobe pad, then this has the equivelent functionality of legacy receiver enable delay. When applied to a CSR for a DQ pad, then this has the equivelent functionality of legacy read DQS delay.

D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]81_dct[0] Tx Delay

Cold reset: 0000_0000h. See 2.9.4.1 for chiplet to pad mapping and see 2.9.4 for pad to pin mapping.

Table 164: Index Mapping for D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]81_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][11:8]	[3:0]	DIMM
D18F2x98_dct[0][11:8]	Fh	DIMM[3:0]
D18F2x98_dct[0][15:12]	[3:0]	Timing Group
D18F2x98_dct[0][15:12]	Fh	Timing Group [3:0]
D18F2x98_dct[0][23:20]	[7:0]	DBYTE chiplet
D18F2x98_dct[0][23:20]	8	Reserved
D18F2x98_dct[0][23:20]	Fh	DBYTE chiplet [7:0]
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:8	Reserved.
7:0	**TxDly: Tx delay**. Read-write. BIOS: See 2.9.9.2.10. Specifies the time that the DATA (includes both DQ and DQS/WCK) chip signals are delayed from the default setup time, in increments of 1/32 UI. When applied to a CSR for a write strobe pad, then this has the equivelent functionality of legacy write DQS delay. When applied to a CSR for a DQ pad, then this has the equivelent functionality of legacy write data delay.

D18F2x9C_x0002_0000_dct[0] PLL MemoryPstate0

Cold reset: 0000_0400h.

Table 165: Index Mapping for PllMultDiv Value Definition

Bits	Data Rate
0603h	667 MT/s
0D01h	1333 MT/s
1000h	1600 MT/s
1203h	1866 MT/s
1501h	2133 MT/s
1800h	2400 MT/s
Note: “Memory-Clock” frequency is one-half of the data rate.	

Bits	Description
31:16	Reserved.
15:0	**PllMultDiv: PLL multiplier divider**. Read-write. BIOS: 2.9.9.2.3. Specifies the DRAM bus data rate for Memory P-state 0, by way of PHY PLL multiplier and divider. See Table 165.

D18F2x9C_x0002_0001_dct[0] PLL MemoryPstate1

Cold reset: 0000_0400h.

Bits	Description
31:16	Reserved.
15:0	**PllMultDiv: PLL multiplier divider**. Read-write. BIOS: 2.9.9.2.3. Specifies the DRAM bus data rate for memory P-state 1 by way of PHY PLL multiplier and divider. See Table 165.

D18F2x9C_x0002_0004_dct[0] Mailbox Protocol Shadow

Cold reset: 0000_0007h.

Bits	Description
31:3	Reserved.
2	**Us2Rdy: Upstream 2 Ready**. Read-only. This bit is a read-only copy of D18F2x9C_x0002_0035_dct[0][Rdy].
1	**UsRdy: Upstream Ready**. Read-only. This bit is a read-only copy of D18F2x9C_x0002_0033_dct[0][Rdy].
0	Reserved.

D18F2x9C_x0002_000B_dct[0] Power State Command

Cold reset: 0000_0004h.

Bits	Description
31:16	Reserved.
15:14	**CmdType: Command Type**. Read-write. Bits: Description 00b: PHY P-State Command 01b: NB-Pstate Update Command 11b-10b: Reserved
13:11	Reserved.
10:8	**NbPstate: Nb Pstate**. Read-write.
7:5	Reserved.
4	**EnterS3: Enter S3**. Read-write. 1=PHY removes the MemReset control from DCT interface in preparation of S3 state transition. Only valid with LP2 command. Valid when CmdType == 0.
3	Reserved.
2:0	**PhyPowerState: PHY Power State**. Read-write. Specifies the PHY power state to transition to when this register is written. Valid when CmdType == 0. Bits: Description 000b: PS0 001b: PS1 011b-010b: Reserved 100b: LP2 111b-101b: Reserved

D18F2x9C_x0002_000E_dct[0] Global Control

Cold reset: 0000_0002h.

Bits	Description
31:2	Reserved.
1	**PhyDisable: PHY Disable**. Read-write. BIOS: 2.9.9.2. 1=PHY is disabled. 0=PHY is enabled.
0	**D4_Mode**. Read-write. 1=DDR4 mode. 0=DDR3 mode.

D18F2x9C_x0002_0015_dct[0] Vref Byte

Cold reset: 0000_0000h.

Bits	Description
31:4	Reserved.
3:0	**VrefFilt: Vref filter**. Read-write. BIOS: 0. This field adjusts noise coupling on to VrefOut and adjusts the input resistance.

D18F2x9C_x0002_001A_dct[0] ByteDbgCtrl

Cold reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:8	**DbDbgSelByte1**. Read-write.
7:0	**DbDbgSelByte0**. Read-write.

D18F2x9C_x0002_0032_dct[0] Upstream Mailbox 1 Message

Cold reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**Message: Message.** Read-write. This field specifies the encoded message received.

D18F2x9C_x0002_0033_dct[0] Upstream Mailbox 1 Protocol

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**Rdy: Ready.** Read-write. 1=The Upstream Mailbox 1 is ready for a data transfer from the PMU to the mailbox; the PMU may write to D18F2x9C_x0002_0032_dct[0][Message]. 0=PMU may not write to D18F2x9C_x0002_0032_dct[0][Message].

D18F2x9C_x0002_0034_dct[0] Upstream Mailbox 2 Message

Cold reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**Message: Message.** Read-write. This field specifies the data received.

D18F2x9C_x0002_0035_dct[0] Upstream Mailbox 2 Protocol

Cold reset: 0000_0001h.

Bits	Description
31:1	Reserved.
0	**Rdy: Ready.** Read-write. 1=The Upstream Mailbox 2 is ready for a data transfer from the PMU to the mailbox; the PMU may write to D18F2x9C_x0002_0034_dct[0][Message]. 0=PMU may not write to D18F2x9C_x0002_0034_dct[0][Message].

D18F2x9C_x0002_005A_dct[0] D3_MERR_RCVR_CNTRL

Cold reset: 0000_0681h.

Bits	Description
31:15	Reserved.
14:11	**D3MERR_PuEn**. Read-write.
10	**D3MERR_DisableVal**. Read-write. Specifes the forwarded value when D3MERR_RxEn=0.
9	**D3MERR_AfeEn**. Read-write.
8:1	**D3MERR_RefSel**. Read-write. Specifies the Vref level.
0	**D3MERR_RxEn**. Read-write. 1=Enable receiver for MERR_L pad. 0=Disable the receiver and D3EVNT_DisableVal state is forwarded.

D18F2x9C_x0002_005B_dct[0] D3_EVNT_RCVR_CNTRL

Cold reset: 0000_0681h.

Bits	Description
31:15	Reserved.
14:11	**D3EVNT_PuEn**. Read-write.
10	**D3EVNT_DisableVal**. Read-write.Specifes the forwarded value when D3EVNT_RxEn=0.
9	**D3EVNT_AfeEn**. Read-write.
8:1	**D3EVNT_RefSel**. Read-write. Specifies the Vref level.
0	**D3EVNT_RxEn**. Read-write. 1=Enable receiver for EVENT_L pad. 0=Disable the receiver and D3EVNT_DisableVal state is forwarded.

D18F2x9C_x0002_005D_dct[0] Misc5

Cold reset: 0000_0000h.

Bits	Description
31:2	Reserved.
1	**ForceHClk.** Read-write.
0	Reserved.

D18F2x9C_x0002_005F_dct[0] Misc PHY Status

Cold reset: 0000_000Xh.

Bits	Description
31:2	Reserved.
1	**PORMemReset: POR MemReset.** Read-only. 1=Previous cold reset was a power-up event (accompanied by ramping VDDIO). 0=Previous cold reset was associated with S3 (VDDIO was already supplied).
0	**DctSane: Dct Sane.** Read-only; Updated-by-hardware. 1=Software provides the logic state of Memreset_L to drive as specified by D18F2x9C_x0002_0060_dct[0][MemReset]. 0=MemReset_L logic state is provided by the complement of PORMemReset. This bit is set by hardware after the first write to D18F2x9C_x0002_000B_dct[0].

D18F2x9C_x0002_0060_dct[0] Memreset Control

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**MemReset_L.** Read-write. BIOS: See 2.9.9.5. 1=MemReset_L pin is driven inactive when D18F2x9C_x0002_005F_dct[0][DctSane] == 1. 0=MemReset_L pin is driven active when D18F2x9C_x0002_005F_dct[0][DctSane] == 1.

D18F2x9C_x0[3,1:0]02_0080_dct[0] PMU CLK Divider

Cold reset: 0000_0008h.

Table 166: Index Mapping for D18F2x9C_x0[3,1:0]02_0080_dct[0]

Address Bits	Valid Values	Name
D18F2x98_dct[0][27:24]	[1:0]	Memory Pstate
D18F2x98_dct[0][27:24]	3h	Memory Pstate [1:0]

Bits	Description
31:4	Reserved.
3:0	**PMUClkDiv: PMU CLK divider.** Read-write. BIOS: 2.9.9.2.3. Specifies the divider from PCLK to the PMU input clock. PCLK frequency is the same as the data rate (See D18F2x9C_x0002_0000_dct[0][PllMultDiv] and D18F2x9C_x0002_0001_dct[0][PllMultDiv]). This must be programmed so the PMU operates no faster than 533 MHz. Software should program this to the smallest divider which meets this condition for best training results. For example, if the data rate is 5Gb/s (PllMultDiv == 3200h) then PMUClkDiv=Divide by 10 (the PMU runs at 500 MHz). If for example, the data rate is 667Mb/s (PllMultDiv == 0603h) then PMUClkDiv=Divide by 2 (the PMU runs at 333 MHz). When training is complete and the PMU is in reset, it is recommended that BIOS program this to PMU CLK = CPU RefClk. Bits — Description 0000b — Divide by 1 0001b — Divide by 2 0010b — Divide by 4 0011b — Divide by 6 0100b — Divide by 8 0101b — Divide by 10 0110b — Reserved 0111b — PMU CLK turned off 1111b-1000b — PMU CLK = Refclk

D18F2x9C_x0002_0087_dct[0] CalAndPllConfig

Cold reset: 0000_0000h.

Bits	Description
31:5	Reserved.
4	**EnUcCfgClkDurTrain**. Read-write.
3	**AlwaysEnCfgClk**. Read-write.
2	**DisablePllPdReg**. Read-write.
1	**DisAutoComp: disable auto compensation**. Read-write. BIOS: 2.9.9.2.5. 1=Disable automatic updates of Tx and POdt targets.
0	**DisPredriverCal: disable predriver calibration**. Read-write. BIOS: 2.9.9.2.5. 1=Disable automatic updates of predriver targets. 0=Enable automatic updates of predriver targets.

D18F2x9C_x0002_0088_dct[0] CalRate

Cold reset: 0000_0083h.

Bits	Description
31:9	Reserved.
8	**CalOdtNeverLock: Calibration init Odt no-lock mode**. Read-write. BIOS: 2.9.9.2.5.
7	**CalInitMode: Calibration init mode**. Read-write. BIOS: 2.9.9.2.5.
6	Reserved.
5	**CalOnce: Calibration run one time**. Read-write. BIOS: 2.9.9.2.5. 1=Run one time. 0=Run continuously.
4	**CalRun: Calibration run**. Read-write. BIOS: 2.9.9.2.5.
3:0	**CalInterval: Calibration interval**. Read-write. BIOS: 2.9.9.2.5.

D18F2x9C_x0002_0089_dct[0] PllLockTime

Cold reset: 0000_00C8h.

Bits	Description
31:10	Reserved.
9:0	**PllLockTime: PLL lock time**. Read-write. BIOS: 2.9.9.2.3. Specifies the number of 10 ns periods the PHY waits for PLLs to lock during a frequency change.

D18F2x9C_x0002_008A_dct[0] PllDllLockStatus

Cold reset: 0000_0000h.

Bits	Description
31:2	Reserved.
1	**DllLocked**. Read-only; Updated-by-hardware.
0	**PllLocked**. Read-only; Updated-by-hardware.

D18F2x9C_x0002_008B_dct[0] Pstate

Cold reset: 0000_0002h.

Bits	Description
31:2	Reserved.
1:0	**NB_Pstate.** Read-write, Updated-by-hardware. Bits: Description 11b-00b: <NB_Pstate>

D18F2x9C_x0002_008E_dct[0] MERR_status

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**MemErrH_sticky**. Read-only.

D18F2x9C_x0002_0093_dct[0] PllRegWaitTime

Cold reset: 0000_0023h.

Bits	Description
31:8	Reserved.
7:0	**PllRegWaitTime: PLL regulator wait time.** Read-write. BIOS: 2.9.9.2.3. Specifies the number of 10 ns periods the PHY waits for the PLL to become stable when coming out of PLL regulator off power down mode.

D18F2x9C_x0002_0097_dct[0] CalBusy

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**CalBusy: Calibration busy.** Read-only.

D18F2x9C_x0002_0098_dct[0] CalMisc2

Cold reset: 0000_0204h.

Bits	Description
31:14	Reserved.
13	**CalD4D3**. Read-write. BIOS: 0. 1=Reserved. 0=Calibrate DDR3 and DDR4.
12:11	Reserved.
10	**CalOdtFinishAsap**. Read-write.
9:3	**CalOdtMaxIteration**. Read-write.
2:0	**CalNumVotes**. Read-write.

D18F2x9C_x0002_0099_dct[0] PMU Reset

Cold reset: 0000_0001h.

Bits	Description
31:7	Reserved.
6	**SRAM_SD: PMU SRAM Shutdown.** Read-write. BIOS: 2.9.9.9. 1=Power down PMU SRAM.
5:4	Reserved.

3	**PmuReset: PMU Reset.** Read-write. BIOS: 2.9.9.2. 1=Places the PHY microcontroller unit (PMU) in reset. 0=Starts clocks and removes the reset signal to the PMU.
2:1	Reserved.
0	**PmuStall: PMU Stall.** Read-write. BIOS: 2.9.9.2. 1=Places the PHY microcontroller unit (PMU) in a clock gated stall state. 0=Starts clocks and resume execution at current instruction pointer.

D18F2x9C_x0002_009B_dct[0] CalVRefs

Cold reset: 0000_EA02h.

Bits	Description
31:16	Reserved.
15:12	**CalVrefSp.** Read-write.
11:8	**CalVref06.** Read-write.
7:4	**CalVref05.** Read-write.
3:0	**CalVref04.** Read-write.

D18F2x9C_x0002_00B2_dct[0] VrefInGlobal

Cold reset: 0000_0003h.

Bits	Description
31:10	Reserved.
9:3	**GlobalVrefInDAC.** Read-write.
2:0	**GlobalVrefInSel.** Read-write. BIOS: 1.

D18F2x[B,0]9C_x0005_[5FFF:4000]_dct[0] PMU IC SRAM

Cold reset: 0000_0000h.
This is a word-addressable address space.
Software must write a pair of words into PMU SRAM, starting with an numbered index, in order for a write to be properly latched (e.g., writing to SRAM index 0 and then index 1 results in two data words being written to SRAM); writing to index 0 only or index 1 only results in no data being written to SRAM. If writing an even numbered word-sized block of SRAM then no additional writes are necessary. If writing with random access and software loses track of how many words were written, then it should assume the write was not latched and re-write utilizing a pair of accesses as stated above. It is not recommended to interrupt two consecutive writes with an intervening read operation.

Bits	Description
31:16	Reserved.
15:0	**PMUFirmwareSRAM: PMU Firmware SRAM**. Read-write. BIOS: 2.9.9.2.9.

D18F2x9C_x0005_[0FFF:0000]_dct[0] PMU SRAM Message Block

Cold reset: 0000_0000h.
This is a word-addressable address space. The lower 256 bytes of SRAM in the data portion of the address map is the SRAMMsgBlk. The remaining portion of the SRAM is used by the system.

Software must write a pair of words into PMU SRAM, starting with an even numbered index, in order for a write to be properly latched (e.g., writing to SRAM index 0 and then index 1 results in two data words being written to SRAM); writing to only an even index will result in no data written to SRAM; writing to only an odd index results in arbitrary data written to the memory addressed by the corresponding even index (destroying the prior contents) but the odd index is written with the intended data. If writing an even numbered word-sized block of SRAM aligned on an even index then no additional writes are necessary. If writing with random access and software loses track of how many words were written, then it should assume the write was not latched and re-write utilizing a pair of accesses as stated above. It is not recommended to interrupt two consecutive writes with an intervening read operation.

Bits	Description
31:16	Reserved.
15:0	**SRAMMsgBlk: SRAMMsgBlk**. Read-write. BIOS: 2.9.9.2.9.

D18F2x9C_x00[F,1:0]4_00E[7:0]_dct[0] OdtCtrl

Cold reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:4	**OdtRdPatCs: Odt read pattern Cs[7:0].** Read-write. BIOS: See D18F2x[234:230]_dct[0]. This register specifies the ODT[3:0] pin pattern during a read to chipselect [7:0] for PMU training.
3:0	**OdtWrPatCs: Odt write pattern Cs[7:0].** Read-write. BIOS: See D18F2x[23C:238]_dct[0]. This register specifies the ODT[3:0] pin pattern during a write to chipselect [7:0] for PMU training.

D18F2x9C_x00[F,1:0]4_00E8_dct[0] OdtCtrl8

Cold reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:12	**AcsmOdtRdStrtCtrl.** Read-write. ODT RD pulse start time from CAS in MCLK units.
11:8	**AcsmOdtWrStrtCtrl.** Read-write. ODT WR pulse start time from CAS in MCLK units.
7:4	**AcsmOdtRdDurCtrl.** Read-write. ODT RD pulse width in MCLK units.
3:0	**AcsmOdtWrDurCtrl.** Read-write. ODT WR pulse width in MCLK units.

D18F2x9C_x00[F,1:0]4_00FD_dct[0] PHY CKE control

Cold reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:4	**AcsmCkeEnb[3:0]: ACSM CKE enable[3:0]**. Read-write. For each bit, 1=If (AcsmCkeOride == 1) then CKE is asserted active. 0=If (AcsmCkeOride == 1) then CKE is deasserted.
3:0	**AcsmCkeOride[3:0]: ACSM CKE override[3:0]**. Read-write. For each bit, 1=When the DCT is not connected, PMU overrides the ACSM output values, allowing directly programmability with AcsmCkeEnb. 0=When the DCT is not connected, the ACSM determines CKE output values.

D18F2x9C_x0007_0015_dct[0] Lane to CRC Map0

Cold reset: 0000_0000h.

Bits	Description
31:12	Reserved.
11:9	**CrcLaneMap3: Crc lane map 3**. Read-write. Map lane 3 to CRC input.
8:6	**CrcLaneMap2: Crc lane map 2**. Read-write. Map lane 2 to CRC input.
5:3	**CrcLaneMap1: Crc lane map 1**. Read-write. Map lane 1 to CRC input.
2:0	**CrcLaneMap0: Crc lane map 0**. Read-write. Map lane 0 to CRC input.

D18F2x9C_x0007_0016_dct[0] Lane to CRC Map1

Cold reset: 0000_0000h.

Bits	Description
31:12	Reserved.
11:9	**CrcLaneMap7: Crc lane map 7**. Read-write. Map lane 7 to CRC input.
8:6	**CrcLaneMap6: Crc lane map 6**. Read-write. Map lane 6 to CRC input.
5:3	**CrcLaneMap5: Crc lane map 5**. Read-write. Map lane 5 to CRC input.
2:0	**CrcLaneMap4: Crc lane map 4**. Read-write. Map lane 4 to CRC input.

D18F2xA4 DRAM Controller Temperature Throttle

See 2.9.3 [DCT Configuration Registers].
See 2.9.14 [DRAM On DIMM Thermal Management and Power Capping].

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31:24</td><td>Reserved.</td></tr>
<tr><td>23:20</td><td>BwCapCmdThrottleMode: bandwidth capping command throttle mode. Read-write. Reset: 0. Specifies the command throttle mode when BwCapEn == 1. The DCT throttles commands over a rolling window of 100 clock cycles, maintaining the average throttling as specified by this field.

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0000b</td><td>Command throttling is disabled</td></tr>
<tr><td>0001b</td><td>Throttle commands by 30%</td></tr>
<tr><td>0010b</td><td>Throttle commands by 40%</td></tr>
<tr><td>0011b</td><td>Throttle commands by 50%</td></tr>
<tr><td>0100b</td><td>Throttle commands by 55%</td></tr>
<tr><td>0101b</td><td>Throttle commands by 60%</td></tr>
<tr><td>0110b</td><td>Throttle commands by 65%</td></tr>
<tr><td>0111b</td><td>Throttle commands by 70%</td></tr>
<tr><td>1000b</td><td>Throttle commands by 75%</td></tr>
<tr><td>1001b</td><td>Throttle commands by 80%</td></tr>
<tr><td>1010b</td><td>Throttle commands by 85%</td></tr>
<tr><td>1011b</td><td>Throttle commands by 90%</td></tr>
<tr><td>1100b</td><td>Throttle commands by 95%</td></tr>
<tr><td>1101b</td><td>Reserved</td></tr>
<tr><td>1110b</td><td>Throttle commands as specified by CmdThrottleMode</td></tr>
<tr><td>1111b</td><td>Reserved</td></tr>
</table>
Throttling should not be enabled until after DRAM initialization (D18F2x110[DramEnable] == 1) and training (see 2.9.9.6 [DRAM Training]) are complete.</td></tr>
<tr><td>19:15</td><td>Reserved.</td></tr>
<tr><td>14:12</td><td>CmdThrottleMode: command throttle mode. Read-write. Reset: 0. BIOS: See 2.9.9.4 [DCT Specific Configuration]. Specifies the command throttle mode when ODTSEn == 1 and the EVENT_L pin is asserted. The DCT throttles commands over a rolling window of 100 clock cycles, maintaining the average throttling as specified by this field.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>000b</td><td>Command throttling is disabled.</td></tr>
<tr><td>001b</td><td>Throttle commands by 30%.</td></tr>
<tr><td>010b</td><td>Throttle commands by 50%.</td></tr>
<tr><td>011b</td><td>Throttle commands by 60%.</td></tr>
<tr><td>100b</td><td>Throttle commands by 70%.</td></tr>
<tr><td>101b</td><td>Throttle commands by 80%.</td></tr>
<tr><td>110b</td><td>Throttle commands by 90%.</td></tr>
<tr><td>111b</td><td>Place the DRAM devices in powerdown mode (see D18F2x94_dct[0][PowerDownMode]) when EVENT_L is asserted. This mode is not valid if D18F2x94_dct[0][PowerDownEn] == 0.</td></tr>
</table>
Throttling should not be enabled until after DRAM initialization (D18F2x110[DramEnable] == 1) and training are complete. See also BwCapEn.</td></tr>
</table>

11	**BwCapEn: bandwidth capping enable**. Read-write. Reset: 0. 1=The memory command throttle mode specified by BwCapCmdThrottleMode is applied. This bit is used by software to enable command throttling independent of the state of the EVENT_L pin. If this bit is set, ODTSEn = 1, and the EVENT_L pin is asserted, the larger of the two throttle percentages specified by CmdThrottleMode and BwCapCmdThrottleMode is used.
10:9	Reserved.
8	**ODTSEn: on DIMM temperature sensor enable**. Read-write. Reset: 0. Enables the monitoring of the EVENT_L pin and indicates whether the on DIMM temperature sensors of the DIMMs on a channel are enabled. 0=Disabled. 1=Enabled. While the EVENT_L pin is asserted, the controller (a) doubles the refresh rate if ODTSDoubleRefreshRate == 1, and (b) throttles the address bus utilization as specified by CmdThrottleMode[2:0].
7:1	Reserved.
0	**ODTSDoubleRefreshRate: on DIMM temperature sensor double refresh rate**. Read-write. Reset: 0. 1=Enables a double Tref refresh rate when the EVENT_L pin is asserted. 0=Tref refresh rate when the EVENT_L pin is asserted. Reserved if !ODTSEn. IF (Tref == 7.8 us) THEN ODTSDoubleRefreshRate = 1. ELSE ODTSDoubleRefreshRate = 0. ENDIF.

D18F2xA8_dct[0] DRAM Controller Miscellaneous 2

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31	**PerRankTimingEn: per rank timing enable**. Read-write. Reset: 0. BIOS: 0. Specifies the mapping between chip selects and a set of programmable timing delays. 1=Each chip select is controlled by a set of timing delays. A maximum of 4 chip selects are supported per channel. 0=Each chip select pair is controlled by a set timing delays.
30:29	Reserved.
28	**FastSelfRefEntryDis: fast self refresh entry disable**. Read-write; Same-for-all. Reset: 1. BIOS: IF (D18F5xA0[DisCasBasedOpbdMgmt] == 0), THEN 0. ENDIF. 1=DCT pushes outstanding transactions to memory prior to entering self refresh. 0=DCT enters self refresh immediately unless instructed to push outstanding transactions to memory by D18F2x11C[FlushWrOnStpGnt] or D18F2x1B4[FlushWrOnS3StpGnt].
27:24	Reserved.
23	**OppWrEnhDis: opportunistic write enh disable**. Read-write. Reset: 0. BIOS: 1. Specifies the DCT behavior for opportunistic writes being sent to the DCT not part of a write burst. 1=Disable enhancement logic. 0=Under certain conditions, low priority writes do not arbitrate or participate in auto precharge determination for other commands. See D18F2x11C[DctWrLimit].
22	**PrtlChPDEnhEn: partial channel power down enh enable**. Read-write. Reset: 0. BIOS: 0. Selects the channel idle hysteresis for fast exit/slow exit mode changes when (D18F2x94_dct[0][PowerDownMode] & D18F2x84_dct[0][PchgPDModeSel]). 1=Hysteresis specified by D18F2x244_dct[0][PrtlChPDDynDly]. 0=256 clock hysteresis.
21	**AggrPDEn: aggressive power down enable**. Read-write. Reset: 0. BIOS: 1. 1=The DCT places the DRAM devices in precharge power down mode when pages are closed as specified by D18F2x248_dct[0]_mp[1:0][AggrPDDelay]. 0=The DCT places the DRAM devices in precharge power down mode when pages are closed as specified by D18F2x90_dct[0][DynPageCloseEn].

20	**BankSwap: swap bank address**. Read-write. Reset: 0. BIOS: See 2.9.9.4 [DCT Specific Configuration]. 1=Swap the DRAM bank address bits. The normalized address bits used for bank address are swapped with lower order address bits. If D18F2x114[DctSelBankSwap] == 1 then normalized address bits[10:8] are swapped else normalized address bits[11:9] are swapped. See D18F2x80_dct[0]. This swap happens before D18F2x94_dct[0][BankSwizzleMode] is applied.
19	**PDPhyPSDis: power down phy power save disable**. Read-write. Reset: 0. BIOS: 1. 1=Disable phy clock gating during precharge power down (phy LP1 power state). 0=Enable phy clock gating during precharge power down to save power. BIOS must set this bit prior to setting D18F2x94_dct[0][PowerDownEn] = 1, or before enabling the controller.
18:16	Reserved.
15:8	**MrsCtrlWordCS[7:0]: MRS and control word chip select**. Read-write. Reset: 0. Specifies the target chip selects used for MRS or control word programming. See D18F2x7C_dct[0][SendMrsCmd, SendPDAMrsCmd, SendControlWord]. When used in conjunction with D18F2x7C_dct[0][SendMrsCmd, SendPDAMrsCmd], defined only if (~D18F2x7C_dct[0][EnDramInit] \| ~D18F2x90_dct[0][UnbuffDimm]); otherwise, MRS commands are sent to all chip selects. Bit — Description [0] — CS0 is asserted [1] — CS1 is asserted [2] — CS2 is asserted [3] — CS3 is asserted [4] — CS4 is asserted [5] — CS5 is asserted [6] — CS6 is asserted [7] — CS7 is asserted
7:2	Reserved.
1	**GmcUrgPriMode: GMC urgent priority mode**. Read-write. Reset: 0. 0=GmcUrg elevates the priority of all GMC requests. 1=GmcUrg elevates the priority of only GMC RT (Real Time) request.
0	**EffArbDis: Efficient arbitration disable**. Read-write. Reset: 0. BIOS: 0. 0=The DCT optimizes the arbitration phases to improve performance under certain traffic conditions whenever the NCLK to MEMCLK ratio is less than 2:1. 1=The DCT arbitrates normally, at all NCLK:MEMCLK ratios.

D18F2xAC DRAM Controller Temperature Status

Cold reset: 0000_0000h.

Bits	Description
31:1	Reserved.
0	**MemTempHot0: Memory temperature hot, DCT0**. Read; Write-1-to-clear. 1=The EVENT_L pin was asserted indicating the memory temperature exceeded the normal operating limit; the DCT may be throttling the interface to aid in cooling. See D18F2xA4.

D18F2x110 DRAM Controller Select Low

Reset: 0000_0000h.

Bits	Description
31:11	Reserved.

10	**MemCleared: memory cleared**. Read-only; Updated-by-hardware. 1=Memory has been cleared since the last warm reset. This bit is set by MemClrInit. See MemClrInit.
9	**MemClrBusy: memory clear busy**. Read-only; Updated-by-hardware. 1=The memory clear operation in either of the DCTs is in progress. Reads or writes to DRAM while the memory clear operation is in progress result in undefined behavior.
8	**DramEnable: DRAM enabled**. Read-only. 1=All of the used DCTs are initialized (see 2.9.9.5 [DRAM Device Initialization and Training]) or have exited from self refresh (D18F2x90_dct[0][ExitSelfRef] transitions from 1 to 0).
7:4	Reserved.
3	**MemClrInit: memory clear initialization**. IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. 1=The node writes 0's to all locations of system memory attached to the node and sets the MemCleared bit. The status of the memory clear operation can be determined by reading the MemClrBusy and MemCleared bits. This command is ignored if MemClrBusy == 1 when the command is received. DramEnable must be set before setting MemClrInit. The memory prefetcher must be disabled by setting D18F2x11C[PrefIoDis] and D18F2x11C[PrefCpuDis] before memory clear initialization and then can be re-enabled when MemCleared == 1.
2	**BankSwapAddr8En: Bank swap to address bit[8] enable**. Read-write. BIOS: (D18F2xA8_dct[0][BankSwap] && (D18F2x114[DctSelBankSwap] == 1b)). See D18F2xA8_dct[0][BankSwap].
1:0	Reserved.

D18F2x114 DRAM Controller Select High

IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0000_0000h.

Bits	Description
31:10	Reserved.
9	**DctSelBankSwap: select DRAM bank swap address**. BIOS: 1. See D18F2xA8_dct[0][BankSwap].
8:0	Reserved.

D18F2x118 Memory Controller Configuration Low

Fields in this register (bits[17:0]) indicate priority of request types. Variable priority requests enter the memory controller as medium priority and are promoted to high priority if they have not been serviced in the time specified by MctVarPriCntLmt. This feature may be useful for isochronous IO traffic. If isochronous traffic is specified to be high priority, it may have an adverse effect on the bandwidth and performance of the devices associated with the other types of traffic. However, if isochronous traffic is specified as medium priority, the processor may not meet the isochronous bandwidth and latency requirements. The variable priority allows the memory controller to optimize DRAM transactions until isochronous traffic reaches a time threshold and must be serviced more quickly.

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31:28</td><td>MctVarPriCntLmt: variable priority time limit. Read-write. Reset: 0000b. BIOS: 0001b.
<table>
<tr><th>Bits</th><th>Description</th><th>Bits</th><th>Description</th></tr>
<tr><td>0000b</td><td>80 ns</td><td>1000b</td><td>720 ns</td></tr>
<tr><td>0001b</td><td>160 ns</td><td>1001b</td><td>800 ns</td></tr>
<tr><td>0010b</td><td>240 ns</td><td>1010b</td><td>880 ns</td></tr>
<tr><td>0011b</td><td>320 ns</td><td>1011b</td><td>960 ns</td></tr>
<tr><td>0100b</td><td>400 ns</td><td>1100b</td><td>1040 ns</td></tr>
<tr><td>0101b</td><td>480 ns</td><td>1101b</td><td>1120 ns</td></tr>
<tr><td>0110b</td><td>560 ns</td><td>1110b</td><td>1200 ns</td></tr>
<tr><td>0111b</td><td>640 ns</td><td>1111b</td><td>1280 ns</td></tr>
</table></td></tr>
<tr><td>27</td><td>Reserved.</td></tr>
<tr><td>26:24</td><td>McqHiPriByPassMax: memory controller high priority bypass max. Read-write. Reset: 100b. Specifies the number high-priority operations that are allowed before yielding to medium or low-priority operations. 000b is reserved.</td></tr>
<tr><td>23</td><td>Reserved.</td></tr>
<tr><td>22:20</td><td>McqMedPriByPassMax: memory controller medium bypass low priority max. Read-write. Reset: 100b. Specifies the number of medium-priority operations that are allowed before yielding to low-priority operations. 000b is reserved.</td></tr>
<tr><td>19</td><td>LockDramCfg. Write-1-only. Reset: 0. BIOS: See 2.9.13 [DRAM CC6/PC6 Storage], 2.5.2.2.3.3 [Core C6 (CC6) State].
The following registers are read-only if LockDramCfg == 1; otherwise the access type is specified by the register:
• D18F1xF0 [DRAM Hole Address]
• D18F2x110 [DRAM Controller Select Low]
• D18F2x114 [DRAM Controller Select High]
• D18F4x128[CoreStateSaveDestNode]
• D18F1x[17C:140,7C:40] [DRAM Base/Limit]
• D18F1x120 [DRAM Base System Address]
• D18F1x124 [DRAM Limit System Address]
• D18F2x118[CC6SaveEn]</td></tr>
<tr><td>18</td><td>CC6SaveEn. IF (D18F2x118[LockDramCfg]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 1=CC6 save area is enabled. See 2.5.2.2.7 [BIOS Requirements for Initialization]. BIOS: (D18F4x118[PwrGateEnCstAct0] | D18F4x118[PwrGateEnCstAct1] | D18F4x11C[PwrGateEnCstAct2]).</td></tr>
<tr><td>17:16</td><td>Reserved.</td></tr>
</table>

15:14	**MctPriTrace: trace-mode request priority**. Read-write. Reset: 10b. See: MctPriCpuRd. This must be set to 10b.
13:12	**MctPriIsoc: display refresh read priority**. Read-write. Reset: 10b. See: MctPriCpuRd.
11:10	**MctPriWr: default write priority**. Read-write. Reset: 01b. See: MctPriCpuRd.
9:8	**MctPriDefault: default non-write priority**. Read-write. Reset: 00b. See: MctPriCpuRd.
7:6	**MctPriIsocWr: IO write with the isochronous bit set priority**. Read-write. Reset: 00b. See: MctPriCpuRd. This does not apply to isochronous traffic that is classified as display refresh.
5:4	**MctPriIsocRd: IO read with the isochronous bit set priority**. Read-write. Reset: 10b. See: MctPriCpuRd. This does not apply to isochronous traffic that is classified as display refresh.
3:2	**MctPriCpuWr: CPU write priority**. Read-write. Reset: 01b. See: MctPriCpuRd.
1:0	**MctPriCpuRd: CPU read priority**. Read-write. Reset: 00b. Bits — Description 00b — Medium 01b — Low 10b — High 11b — Variable

D18F2x11C Memory Controller Configuration High

The two main functions of this register are to control write bursting and memory prefetching.

Write bursting. DctWrLimit and MctWrLimit specify how writes may be burst from the MCT into the DCT to improve DRAM efficiency. When the number of writes in the MCT reaches the value specified in MctWrLimit, then they are all burst to the DCTs at once. Prior to reaching the watermark, a limited number of writes can be passed to the DCTs (specified by DctWrLimit), tagged as low priority, for the DCTs to complete when otherwise idle. Rules regarding write bursting:

- Write bursting mode only applies to low-priority writes. Medium and high priority writes are not withheld from the DCTs for write bursting.
- If write bursting is enabled, writes stay in the MCQ until the threshold specified by MctWrLimit is reached.
- Once the threshold is reached, all writes in MCQ are converted to medium priority.
- Any write in MCQ that matches the address of a subsequent access is promoted to either medium priority or the priority of the subsequent access, whichever is higher.
- DctWrLimit only applies to low-priority writes.

Memory prefetching. The MCT prefetcher detects stride patterns in the stream of requests and then, for predictable stride patterns, generates prefetch requests. A stride pattern is a pattern of requests through system memory that are the same number of cachelines apart. The prefetcher supports strides of -4 to +4 cachelines, which can include alternating patterns (e.g., +1, +2, +1, +2), and can prefetch 1, 2, 3, 4, or 5 cachelines ahead, depending on the confidence. In addition, a fixed stride mode (non-alternating) may be used for IO requests which often have fixed stride patterns. This mode bypasses the stride predictor such that CPU-access stride predictions are not adversely affected by IO streams.

The MCT tracks several stride patterns simultaneously. Each of these has a confidence level associated with it that varies as follows:

- Each time a request is received that matches the stride pattern, the confidence level increases by one.
- Each time a request is received within +/- 4 cachelines of the last requested cacheline in the pattern that does not match the pattern, then the confidence level decreases by one.
- When the confidence level reaches the saturation point specified by PrefConfSat, then it no-longer incre-

ments.

Each request that is not within +/- 4 cachelines of the last requested cacheline line of all the stride patterns tracked initiates a new stride pattern by displacing one of the existing least-recently-used stride patterns.

The memory prefetcher uses an adaptive prefetch scheme to adjust the prefetch distance based upon the buffer space available for prefetch request data. The adaptive scheme counts the total number of prefetch requests and the number of prefetch requests that cannot return data because of buffer availability. After every 16 prefetch requests, the prefetcher uses the following rules to adjust the prefetch distance:

- If the ratio of prefetch requests that cannot return data to total prefetch requests is greater than or equal to D18F2x1B0[AdapPrefMissRatio] then the prefetch distance is reduced by D18F2x1B0[AdapPrefNegativeStep].
- If the ratio of prefetch requests that cannot return data to total prefetch requests is less than D18F2x1B0[AdapPrefMissRatio] then the prefetch distance is increased by D18F2x1B0[AdapPrefPositiveStep].
- If the adjusted prefetch distance is greater than the prefetch distance defined for the current confidence level, the prefetch distance for the current confidence level is used.

The adaptive prefetch scheme supports fractional prefetch distances by alternating between two whole number prefetch distances. For example a prefetch distance of 1.25 causes a prefetch distance sequence of: 1, 1, 1, 2, 1, 1, 1, 2.

Bits	Description
31	Reserved.
30	**FlushWr: flush writes command**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Setting this bit causes write bursting to be canceled and all outstanding writes to be flushed to DRAM. This bit is cleared when all writes are flushed to DRAM.
29	**FlushWrOnStpGnt: flush writes on stop-grant**. Read-write. Reset: 0. 1=Causes write bursting to be canceled and all outstanding writes to be flushed to DRAM when in the stop-grant state.
28	Reserved.
27:25	**PrefThreeConf: prefetch three-ahead confidence**. Read-write. Reset: 100b. BIOS: 110b. Confidence level required in order to prefetch three cachelines ahead (same encoding as PrefTwoConf below).
24:22	**PrefTwoConf: prefetch two-ahead confidence**. Read-write. Reset: 011b. BIOS: 011b. Confidence level required in order to prefetch two cachelines ahead. Bits — Description 000b — 0 110b-001b — [PrefTwoConf*2] 111b — 14
21:20	**PrefOneConf: prefetch one-ahead confidence**. Read-write. Reset: 10b. BIOS: 10b. Confidence level required in order to prefetch one ahead (0 through 3).
19:18	**PrefConfSat: prefetch confidence saturation**. Read-write. Reset: 00b. BIOS: 00b. Specifies the point at which prefetch confidence level saturates and stops incrementing. Bits — Description 00b — 15 01b — 7 10b — 3 11b — Reserved

17:16	**PrefFixDist: prefetch fixed stride distance**. Read-write. Reset: 00b. Specifies the distance to prefetch ahead if in fixed stride mode. 00b=1 cacheline; 01b=2 cachelines; 10b=3 cachelines; 11b=4 cachelines.
15	**PrefFixStrideEn: prefetch fixed stride enable**. Read-write. Reset: 0. 1=The prefetch stride for all requests (CPU and IO) is fixed (non-alternating).
14	**PrefIoFixStrideEn: Prefetch IO fixed stride enable**. Read-write. Reset: 0. 1=The prefetch stride for IO requests is fixed (non-alternating).
13	**PrefIoDis: prefetch IO-access disable**. Read-write. Reset: 1. BIOS: 0. 1=Disables IO requests from triggering prefetch requests.
12	**PrefCpuDis: prefetch CPU-access disable**. Read-write. Reset: 1. BIOS: 0. 1=Disables CPU requests from triggering prefetch requests.
11:7	**MctPrefReqLimit: memory controller prefetch request limit**. Read-write. Reset: 1Eh. BIOS: 1Dh. Specifies the maximum number of outstanding prefetch requests allowed. See D18F3x78 for restrictions on this field.
6:2	**MctWrLimit: memory controller write-burst limit**. Read-write. Reset: 1Fh. BIOS: 14h. Specifies the number of writes in the memory controller queue before they are burst into the DCTs. Bits — Description 00h — 32 1Dh-01h — [32-MctWrLimit] 1Eh — 2 1Fh — Write bursting disabled
1:0	**DctWrLimit: DRAM controller write limit**. Read-write. Reset: 00b. BIOS: 01b. Specifies the maximum number of writes allowed in the DCT queue when write bursting is enabled, prior to when the number of writes in MCQ exceeds the watermark specified by MctWrLimit. Bits — Description 00b — 0 01b — 2 10b — 4 11b — 8

D18F2x1[7C:40]_dct[0] DQ Mapping

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Table 167: Register Mapping for D18F2x1[7C:40]_dct[0]

Register	Function
D18F2x14[C:0]	DIMM 0
D18F2x15[C:0]	DIMM 1
D18F2x16[C:0]	DIMM 2
D18F2x17[C:0]	DIMM 3

Table 168: Field Mapping for D18F2x1[7C:40]_dct[0]

Register	Bits					
	31:30	29:24	23:18	17:12	11:6	5:0
D18F2x1[7:4]0_dct[0]	Reserved	DQ[19:16]	DQ[15:12]	DQ[11:8]	DQ[7:4]	DQ[3:0]
D18F2x1[7:4]4_dct[0]	Reserved	DQ[39:36]	DQ[35:32]	DQ[31:28]	DQ[27:24]	DQ[23:20]
D18F2x1[7:4]8_dct[0]	Reserved	DQ[59:56]	DQ[55:52]	DQ[51:48]	DQ[47:44]	DQ[43:40]
D18F2x1[7:4]C_dct[0]	Reserved	Reserved	Reserved	DQ[71:68]	DQ[67:64]	DQ[63:60]

Bits	Description
31:30	Reserved.
29:24	**DQMap: DQ map**. Read-write. See: D18F2x1[7C:40]_dct[0][5:0].
23:18	**DQMap: DQ map**. Read-write. See: D18F2x1[7C:40]_dct[0][5:0].
17:12	**DQMap: DQ map**. Read-write. See: D18F2x1[7C:40]_dct[0][5:0].
11:6	**DQMap: DQ map**. Read-write. See: D18F2x1[7C:40]_dct[0][5:0].
5:0	**DQMap: DQ map**. Read-write. Reserved if !(Ddr4Mode & WrCrcEn). Programmed per DIMM SPD map in Ddr4Mode and used for CRC computation.

D18F2x1A8_dct[0] PSM Index

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:0	**PsmIndex: PSM index**. Read-write.

D18F2x1AC_dct[0] PSM Index

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:0	**PsmData: PSM data**. Read-write.

D18F2x1B0 Extended Memory Controller Configuration Low

The main function of this register is to control the memory prefetcher. See D18F2x11C [Memory Controller Configuration High] about the adaptive prefetch scheme.

Table 169: BIOS Recommendations for D18F2x1B[4:0]

Condition	D18F2x1B0	D18F2x1B4	
DdrRate	DcqBwThrotWm	DcqBwThrotWm1	DcqBwThrotWm2
667	0h	3h	4h
800	0h	3h	5h
1066	0h	4h	6h
1333	0h	5h	8h

Table 169: BIOS Recommendations for D18F2x1B[4:0] (Continued)

Condition	D18F2x1B0	D18F2x1B4	
DdrRate	DcqBwThrotWm	DcqBwThrotWm1	DcqBwThrotWm2
1600	0h	6h	9h
1866	0h	7h	Ah
2133	0h	8h	Ch

Bits	Description
31:28	**DcqBwThrotWm: dcq bandwidth throttle watermark**. Read-write. Reset: 3h. BIOS: Table 169. Specifies the number of outstanding DRAM read requests before new DRAM prefetch requests and speculative prefetch requests are throttled. 0h=Throttling is disabled. Legal values are 0h through Ch. Programming this field to a non-zero value disables D18F2x1B4[DcqBwThrotWm1, DcqBwThrotWm2].
27:25	**PrefFiveConf: prefetch five-ahead confidence**. Read-write. Reset: 110b. BIOS: 111b. Confidence level required in order to prefetch five cachelines ahead. Bits — Description 000b — 0 110b-001b — [PrefFiveConf*2] 111b — 14
24:22	**PrefFourConf: prefetch four-ahead confidence**. Read-write. Reset: 101b. BIOS: 111b. Confidence level required in order to prefetch four cachelines ahead. Bits — Description 000b — 0 110b-001b — [PrefFourConf*2] 111b — 14
21	Reserved.
20	**DblPrefEn: double prefetch enable**. Read-write. Reset: 0. 1=The memory prefetcher only generates prefetch requests when it is able to generate a pair of prefetch requests to consecutive cache lines.
19:18	**MctPriIOWr: mct priority for IO writes**. Read-write. Reset: 01b. Sets priority for IO writes from the legacy Onion link. Bits — Description 00b — Medium 01b — Low 10b — High 11b — Variable
17:13	Reserved. Reset: 1_1100b.
12	**EnSplitDctLimits: split DCT write limits enable**. Read-write. Reset: 0. BIOS: 1. 1=The number of writes specified by D18F2x11C[DctWrLimit, MctWrLimit] is per DCT. 0=The number of writes specified by D18F2x11C[DctWrLimit, MctWrLimit] is for the even[0,2] or odd[1,3] DCT channels. 0=The number of writes specified by D18F2x11C[DctWrLimit, MctWrLimit] is total writes independent of DCT. Setting this bit also affects the encoding of D18F2x11C[DctWrLimit].
11	**DisIoCohPref: disable coherent prefetched for IO**. Read-write. Reset: 0. BIOS: 1. 1=Probes are not generated for prefetches generated for reads from IO devices.

10:8	**CohPrefPrbLmt: coherent prefetch probe limit**. Read-write. Reset: 000b. BIOS: 000b. Specifies the maximum number of probes that can be outstanding for memory prefetch requests. Bits Description 000b Probing disabled for memory prefetch requests 111b-001b Reserved.
7:6	**MctPriIORd: mct priority for IO reads**. Read-write. Reset: 00b. Sets priority for IO reads from the legacy link. Bits Description 00b Medium 01b Low 10b High 11b Variable
5:4	**AdapPrefNegativeStep: adaptive prefetch negative step**. Read-write. Reset: 00b. Specifies the step size that the adaptive prefetch scheme uses when decreasing the prefetch distance. Bits Description 00b 2/16 01b 4/16 10b 8/16 11b 16/16
3:2	**AdapPrefPositiveStep: adaptive prefetch positive step**. Read-write. Reset: 00b. Specifies the step size that the adaptive prefetch scheme uses when increasing the prefetch distance. Bits Description 00b 1/16 01b 2/16 10b 4/16 11b 8/16
1:0	**AdapPrefMissRatio: adaptive prefetch miss ratio**. Read-write. Reset: 00b. BIOS: 01b. Specifies the ratio of prefetch requests that do not have data buffer available to the total number of prefetch requests at which the adaptive prefetch scheme begins decreasing the prefetch distance. Bits Description 00b 1/16 01b 2/16 10b 4/16 11b 8/16

D18F2x1B4 Extended Memory Controller Configuration High Register

Bits	Description
31	**FlushOnMmioWrEn: flush on mmio write enable**. Read-write. Reset: 0. 1=Any CPU-sourced MMIO write that matches D18F1x[2CC:2A0,1CC:180,BC:80] causes the memory controller data buffers to be flushed to memory.
30:28	**S3SmafId: S3 SMAF id**. Read-write. Reset: 100b. SMAF encoding of D18F3x[84:80] corresponding to the ACPI S3 state when FlushWrOnS3StpGnt == 1. Reserved when FlushWrOnS3StpGnt == 0.

27	**FlushWrOnS3StpGnt: flush write on S3 stop grant**. Read-write. Reset: 0. BIOS: IF (D18F5xA0[DisCasBasedOpbdMgmt] == 0), THEN 1. ENDIF. 1=Write bursting is canceled and all outstanding writes are flushed to DRAM when in the stop-grant state and the SMAF code is equal to S3SmafId, indicating entry into the ACPI S3 state. See D18F2xA8_dct[0][FastSelfRefEntryDis], D18F2x11C[FlushWrOnStpGnt].
26	**EnSplitMctDatBuffers: enable split MCT data buffers**. Read-only. Reset: 1. 1=Enable resource allocation into the split buffer resources. BIOS must program this bit before any DRAM memory accesses are issued from the processor.
25	**SmuCfgLock: SMU configuration lock**. Read-write; Updated-by-hardware. Reset: 0. This field should never be cleared by software. The following registers are read-only if SmuCfgLock == 1; otherwise the access type is specified by the register: • D18F4x15C [Core Performance Boost Control] • D18F5x170 [Northbridge P-state Control]
24:23	Reserved.
22	**SpecPrefDisWm1: speculative prefetch disable watermark 1**. Read-write. Reset: 0. 0=Disable speculative prefetches at the DcqBwThrotWm2 limit. 1=Disable speculative prefetches at the DcqBwThrotWm1 limit. See also D18F2x1B0[SpecPrefDis].
21	**RegionAlloWm2: region prefetch allocate watermark 2**. Read-write. Reset: 0. See DemandAlloWm2.
20	**RegionPropWm2: region prefetch propagate watermark 2**. Read-write. Reset: 0. See DemandPropWm2.
19	**StrideAlloWm2: stride prefetch allocate watermark 2**. Read-write. Reset: 1. See DemandAlloWm2.
18	**StridePropWm2: stride prefetch propagate watermark 2**. Read-write. Reset: 1. See DemandPropWm2.
17	**DemandAlloWm2: demand request allocate watermark 2**. Read-write. Reset: 1. Specifies the behavior from the DcqBwThrotWm1 limit to the DcqBwThrotWm2 limit. 0=Requests do not allocate a new entry. 1=Requests allocate a new entry; defined only if (DemandAlloWm1 & DemandPropWm2).
16	**DemandPropWm2: demand request propagate watermark 2**. Read-write. Reset: 1. Specifies the behavior from the DcqBwThrotWm1 limit to the DcqBwThrotWm2 limit. 0=Requests do not update existing entries. 1=Requests update existing entries; defined only if (DemandPropWm1 == 1).
15	**RegionAlloWm1: region prefetch allocate watermark 1**. Read-write. Reset: 0. See DemandAlloWm1.
14	**RegionPropWm1: region prefetch propagate watermark 1**. Read-write. Reset: 1. See DemandPropWm1.
13	**StrideAlloWm1: stride prefetch allocate watermark 1**. Read-write. Reset: 1. See DemandAlloWm1.
12	**StridePropWm1: stride prefetch propagate watermark 1**. Read-write. Reset: 1. See DemandPropWm1.
11	**DemandAlloWm1: demand request allocate watermark 1**. Read-write. Reset: 1. Specifies the behavior prior to the DcqBwThrotWm1 limit. 0=Requests do not allocate a new entry. 1=Requests allocate a new entry; defined only if (DemandPropWm1 == 1).
10	**DemandPropWm1: demand request propagate watermark 1**. Read-write. Reset: 1. Specifies the behavior prior to the DcqBwThrotWm1 limit. 0=Requests do not update existing entries. 1=Requests update existing entries.

9:5	**DcqBwThrotWm2: DCQ bandwidth throttle watermark 2**. Read-write. Reset: 06h. BIOS: Table 169. Specifies a prefetch throttling watermark based on the number of outstanding DRAM read requests. This field is reserved when D18F2x1B0[DcqBwThrotWm] != 0. When throttling is enabled, if the number of outstanding DRAM read requests exceeds DcqBwThrotWm2 both request allocate and propagate are blocked and new prefetches are disabled. When throttling is enabled, DcqBwThrotWm2 should be programmed to a value greater than DcqBwThrotWm1. 0h=Throttling is disabled. Legal values are 0h through 18h.
4:0	**DcqBwThrotWm1: DCQ bandwidth throttle watermark 1**. Read-write. Reset: 03h. BIOS: Table 169. Specifies a prefetch throttling watermark based on the number of outstanding DRAM read requests. This field is reserved when D18F2x1B0[DcqBwThrotWm] != 0. 0h=Throttling is disabled. Legal values are 0h through 18h.

D18F2x1B8_dct[0] DRAM ZQ to CS Map

Reset: 0000_0A05h. See 2.9.3 [DCT Configuration Registers]. BIOS: 0000_0A05h & {0x0000, 2{0000b, CS3.CSEnable|CS3.TestFail, CS2.Enable|CS2.TestFail, CS1.Enable|CS1.TestFail, CS0.Enable|CS0.Test-Fail}}.

Table 170: Field Mapping for D18F2x1B8_dct[0]

Register	Bits			
	31:24	23:16	15:8	7:0
D18F2x1B8_dct[0]	ZQ3	ZQ2	ZQ1	ZQ0

Bits	Description
31:24	**CSMapZQ: CS map ZQ**. See: D18F2x1B8_dct[0][7:0].
23:16	**CSMapZQ: CS map ZQ**. See: D18F2x1B8_dct[0][7:0].
15:8	**CSMapZQ: CS map ZQ**. See: D18F2x1B8_dct[0][7:0].
7:0	**CSMapZQ: CS map ZQ**. Read-write. Defines a CS to ZQ command sequence relationship. This register may be configured as one, two, or four groups of ZQ commands per ZqcsInterval. BIOS must not configure ZQ commands to unpopulated chipselects per D18F2x[5C:40]_dct[0]. If Ddr3Mode & UnbuffDimm, only even ZQs may be assigned with even CSes and only odd ZQs may be assigned with odd CSes. 1=This ZQ command is associated with the listed chip select. 0=This ZQ command is not associated with the listed chip select. Bit — Description [0] — CS0 [1] — CS1 [2] — CS2 [3] — CS3 [4] — CS4 [5] — CS5 [6] — CS6 [7] — CS7

D18F2x1BC_dct[0] DRAM CKE to CS Map

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Table 171: Field Mapping for D18F2x1BC_dct[0]

Register	Bits			
	31:24	23:16	15:8	7:0
D18F2x1BC_dct[0]	CKE3	CKE2	CKE1	CKE0

Bits	Description
31:24	**CSMapCKE: CS map CKE**. See: D18F2x1BC_dct[0][7:0].
23:16	**CSMapCKE: CS map CKE**. See: D18F2x1BC_dct[0][7:0].
15:8	**CSMapCKE: CS map CKE**. See: D18F2x1BC_dct[0][7:0].
7:0	**CSMapCKE: CS map CKE**. Read-write. Maps the CS to CKE relationship. 1=This CKE is associated with the listed chip select. 0=This CKE is not associated with the listed chip select. Only even CKEs may be assigned to even CSes. Only odd CKEs may be assigned to odd CSes. Bit — Description [0] — CS0 [1] — CS1 [2] — CS2 [3] — CS3 [4] — CS4 [5] — CS5 [6] — CS6 [7] — CS7

D18F2x1CC_dct[0] Data Scramble Key

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:0	**DataScrambleKey: data scramble key**. Read-write. Specifies the key value used for data scrambling. See D18F2x90_dct[0][DataScrambleEn].

D18F2x1[E8,E0,D8,D0]_dct[0] Performance Event Select Low

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:16	Reserved.
15:8	**UnitMask: event qualification**. Read-write. See MSRC001_024[6,4,2,0][UnitMask].
7:0	**EventSelect[7:0]: event select**. Read-write. See MSRC001_024[6,4,2,0][EventSelect[7:0]].

D18F2x1[EC,E4,DC,D4]_dct[0] Performance Event Select High

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:4	Reserved.
3:0	**EventSelect[11:8]: performance event select**. Read-write. See MSRC001_024[6,4,2,0][EventSelect[11:8]].

D18F2x1F[C:0]_dct[0] DRAM NB P-State Configuration

See 2.9.3 [DCT Configuration Registers].

Table 172: Register Mapping for D18F2x1F[C:0]_dct[0]

Register	Function
D18F2x1F0_dct[0]	NB P-State 0
D18F2x1F4_dct[0]	NB P-State 1
D18F2x1F8_dct[0]	NB P-State 2
D18F2x1FC_dct[0]	NB P-State 3

Bits	Description
31:22	**MaxRdLatency: maximum read latency**. Read-write. Reset: 000h. Specifies the maximum round-trip latency in the system from the processor to the DRAM devices and back. The DRAM controller uses this to help determine when the first two beats of incoming DRAM read data can be safely transferred to the NCLK domain. The time includes the asynchronous and synchronous latencies. Bits: Description 000h: 0 NCLKs 3FEh-001h: <MaxRdLatency> NCLKs 3FFh: 1023 NCLKs
21:19	Reserved.
18:16	**DataTxFifoWrDly: data transmit FIFO write delay**. Read-write. Reset: 0. Specifies the DCT to phy write data FIFO delay. Bits: Description 000b: 0 MEMCLK 001b: 0.5 MEMCLK 010b: 1.0 MEMCLK 011b: 1.5 MEMCLKs 100b: 2.0 MEMCLKs 101b: 2.5 MEMCLKs 110b: 3.0 MEMCLKs 111b: Reserved
15:0	Reserved.

D18F2x200_dct[0]_mp[1:0] DDR3 DRAM Timing 0

Reset: 0F05_0505h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:30	Reserved.
29:24	**Tras: row active strobe**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from an activate command to a precharge command, both to the same chip select bank. Bits: Description 07h-00h: Reserved 36h-08h: <Tras> clocks 3Fh-37h: Reserved
23:21	Reserved.
20:16	**Trp: row precharge time**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from a precharge command to an activate command or auto refresh command, both to the same bank. Bits: Description 04h-00h: Reserved 1Ah-05h: <Trp> clocks 1Fh-1Bh: Reserved
15:13	Reserved.
12:8	**Trcd: RAS to CAS delay**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the time in memory clock cycles from an activate command to a read/write command, both to the same bank. Bits: Description 04h-00h: Reserved 1Ah-05h: <Trcd> clocks 1Fh-1Bh: Reserved
7:5	Reserved.
4:0	**Tcl: CAS latency**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the time in memory clock cycles from the CAS assertion for a read cycle until data return (from the perspective of the DRAM devices). Bits: Description 04h-00h: Reserved 13h-05h: <Tcl> clocks 1Fh-14h: Reserved

D18F2x204_dct[0]_mp[1:0] DDR3 DRAM Timing 1

Reset: 0400_040Bh. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:28	Reserved.

27:24	**Trtp: read CAS to precharge time**. Read-write. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the earliest time in memory clock cycles a page can be closed after having been read. Satisfying this parameter ensures read data is not lost due to a premature precharge. Bits Description 3h-0h Reserved Eh-4h <Trtp> clocks Fh Reserved
23:22	Reserved.
21:16	**FourActWindow: four bank activate window**. Read-write. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the rolling tFAW window in memory clock cycles during which no more than 4 banks in an 8-bank device are activated. Bits Description 00h No tFAW window restriction 05h-01h Reserved 32h-06h [FourActWindow] clocks 3Fh-33h Reserved
15:12	**TrrdL: row to row delay long(or RAS to RAS delay)**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles between activate commands to different chip select banks in the same bank group. Reserved if !Ddr4Mode. D18F2x204_dct[0]_mp[1:0][TrrdL] >= D18F2x204_dct[0]_mp[1:0][Trrd]. Bits Description 0h Reserved Dh-1h <Trrd> clocks Fh-Eh Reserved
11:8	**Trrd: row to row delay (or RAS to RAS delay)**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles between activate commands to different chip select banks. If Ddr4Mode, corresponds to the short parameter and applies between commands to a different bank group. See TrrdL. Bits Description 0h Reserved Dh-1h <Trrd> clocks Fh-Eh Reserved
7	Reserved.
6:0	**Trc: row cycle time**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from and activate command to another activate command or an auto refresh command, all to the same chip select bank. Bits Description 09h-00h Reserved 4Eh-0Ah <Trc> clocks 7Fh-4Fh Reserved

D18F2x208_dct[0]_mp[1:0] DDR3 DRAM Timing 2

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:30	Reserved.
29:20	**Txs: exit self refresh**. Read-write. Reset: 000h. Specifies the time to exit Self Refresh to commands not requiring a locked DLL. Bits: Description 02Fh-000h: Reserved. 2BAh-030h: <Txs> clocks. 3FFh-2BBh: Reserved.
19:16	Reserved.
15:0	**Tref: refresh rate**. Read-write. Reset: 0000h. Specifies the average time between refresh requests. Bits: Description 0000h: Periodic refresh is disabled. FFFFh-0001h: <Tref> clocks.

D18F2x20C_dct[0]_mp[1:0] DDR3 DRAM Timing 3

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31	Reserved.
30:20	**Tdllk: tdllk**. Read-write. Reset: 205h. Specifies the Exit Self Refresh to commands requiring a locked DLL. Bits: Description 7FFh-000h: <Tdllk> clocks.
19:16	Reserved.
15:12	**TwtrL: internal DRAM write to read command delay long**. Read-write. Reset: 4h. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum number of memory clock cycles from a write operation to a read operation, both to the same chip select in the same bank group. This is measured from the rising clock edge following last non-masked data strobe of the write to the rising clock edge of the next read command. Reserved if !Ddr4Mode. D18F2x20C_dct[0]_mp[1:0][TwtrL] >= D18F2x20C_dct[0]_mp[1:0][Twtr]. Bits: Description 1h-0h: Reserved. Eh-2h: <TwtrL> clocks. Fh: Reserved.

11:8	**Twtr: internal DRAM write to read command delay**. Read-write. Reset: 4h. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum number of memory clock cycles from a write operation to a read operation, both to the same chip select. This is measured from the rising clock edge following last non-masked data strobe of the write to the rising clock edge of the next read command. If Ddr4Mode, corresponds to the short parameter and applies between commands to a different bank group. See TwtrL. Bits Description 1h-0h Reserved. Eh-2h <Twtr> clocks. Fh Reserved.
7:5	Reserved.
4:0	**Tcwl: CAS write latency**. Read-write. Reset: 05h. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the number of memory clock cycles from internal write command to first write data in at the DRAM. Bits Description 04h-00h Reserved. 11h-05h <Tcwl> clocks. 1Fh-12h Reserved.

D18F2x214_dct[0]_mp[1:0] DDR3 DRAM Timing 4

Reset: 0001_0202h.

Bits	Description
31:20	Reserved.
19:16	**TwrwrSdSc: write to write timing same DIMM same chip select**. Read-write. BIOS: See 2.9.9.4.1 [DDR Turnaround Parameters]. Specifies the minimum number of cycles from the last clock of virtual CAS of the first write-burst operation to the clock in which CAS is asserted for a following write-burst operation. Bits Description 0h Reserved 1h 1 clock. Ah-2h <TwrwrSdSc> clocks. Bh 11 clocks. Fh-Ch Reserved.
15:12	Reserved.
11:8	**TwrwrSdDc: write to write timing same DIMM different chip select**. See: TwrwrDd.
7:4	Reserved.
3:0	**TwrwrDd: write to write timing different DIMM**. Read-write. BIOS: See 2.9.9.4.1 [DDR Turnaround Parameters]. Specifies the minimum number of cycles from the last clock of virtual CAS of the first write-burst operation to the clock in which CAS is asserted for a following write-burst operation. Bits Description 1h-0h Reserved. Bh-2h <TwrwrDd> clocks. Fh-Ch Reserved.

D18F2x218_dct[0]_mp[1:0] DDR3 DRAM Timing 5

Reset: 0103_0203h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:30	**TrdrdBan: read to read timing ban**. Read-write. BIOS: 01b. Bans the traffic for the specified cases where the number of cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following read-burst operation. Bits: Description 00b: Ban disabled, traffic allowed as specified by TrdrdSdSc, TrdrdSdDc, TrdrdDd. 01b: Ban Trdrd traffic at 2 MEMCLKs. 10b: Ban Trdrd traffic at 2 and 3 MEMCLKs. 11b: Reserved.
29:28	Reserved.
27:24	**TrdrdSdSc: read to read timing same DIMM same chip select**. Read-write. BIOS: See 2.9.9.4.1 [DDR Turnaround Parameters]. Specifies the minimum number of cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following read-burst operation. Bits: Description 0h: Reserved. Bh-1h: <TrdrdSdSc> clocks. Fh-Ch: Reserved.
23:20	Reserved.
19:16	**TrdrdSdDc: read to read timing same DIMM different chip select**. See: TrdrdDd.
15:12	Reserved.
11:8	**Twrrd: write to read DIMM termination turnaround**. Read-write. BIOS: See 2.9.9.4.1 [DDR Turnaround Parameters]. Specifies the minimum number of cycles from the last clock of virtual CAS of the first write-burst operation to the clock in which CAS is asserted for a following read-burst operation, both to different chip selects. Bits: Description 0h: Reserved. Bh-1h: <Twrrd> clocks. Fh-Ch: Reserved.
7:4	Reserved.
3:0	**TrdrdDd: read to read timing different DIMM**. Read-write. BIOS: See 2.9.9.4.1 [DDR Turnaround Parameters]. Specifies the minimum number of cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following read-burst operation. Bits: Description 1h-0h: Reserved. Bh-2h: <TrdrdDd> clocks. Fh-Ch: Reserved.

D18F2x21C_dct[0]_mp[1:0] DDR3 DRAM Timing 6

Reset: 0004_0300h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:28	**TrdrdSdScL: read to read timing same DIMM same chip select long**. Read-write. BIOS: IF Ddr4Mode THEN 2.9.9.3 [SPD ROM-Based Configuration]. ELSE TrdrdSdSc. ENDIF. Specifies the minimum number of cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following read-burst operation in the same bank group. D18F2x218_dct[0]_mp[1:0][TrdrdSdSc] <= D18F2x21C_dct[0]_mp[1:0][TrdrdSdScL] && D18F2x21C_dct[0]_mp[1:0][TrdrdSdScL] <= D18F2x218_dct[0]_mp[1:0][TrdrdSdSc] + 4. Bits: Description 0h: Reserved. Bh-1h: <TrdrdSdScL> clocks. Fh-Ch: Reserved.
27:24	**TwrwrSdScL: write to write timing same DIMM same chip select long**. Read-write. BIOS: IF Ddr4Mode THEN 2.9.9.3 [SPD ROM-Based Configuration]. ELSE TwrwrSdSc. ENDIF. Specifies the minimum number of cycles from the last clock of virtual CAS of a first write-burst operation to the clock in which CAS is asserted for a following write-burst operation in the same bank group. D18F2x214_dct[0]_mp[1:0][TwrwrSdSc] <= D18F2x21C_dct[0]_mp[1:0][TwrwrSdScL] && D18F2x21C_dct[0]_mp[1:0][TwrwrSdScL] <= D18F2x214_dct[0]_mp[1:0][TwrwrSdSc] + 4. Bits: Description 0h: Reserved. Bh-1h: <TwrwrSdScL> clocks. Fh-Ch: Reserved.
23:21	Reserved.
20:16	**TrwtWB: read to write turnaround for opportunistic write bursting**. Read-write. BIOS: TrwtTO + 1. Specifies the minimum number of clock cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following write-burst operation. Bits: Description 02h-00h: Reserved. 1Ch-03h: <TrwtWB> clocks. 1Fh-1Dh: Reserved.
15:13	Reserved.
12:8	**TrwtTO: read to write turnaround**. Read-write. BIOS: See 2.9.9.4.1 [DDR Turnaround Parameters]. Specifies the minimum number of clock cycles from the last clock of virtual CAS of a first read-burst operation to the clock in which CAS is asserted for a following write-burst operation. Bits: Description 01h-00h: Reserved. 1Bh-02h: <TrwtTO> clocks. 1Fh-1Ch: Reserved.
7:0	Reserved.

D18F2x220_dct[0] DDR3 DRAM Timing 7

Reset: 0000_0C04h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:14	Reserved.
13:8	**Tmod: mode register command delay**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from an MRS command to another non-MRS command (excluding NOP and DES), all to the same chip select. Bits: Description 00h: Reserved 3Fh-01h: <Tmod> clocks
7:4	Reserved.
3:0	**Tmrd: mode register command cycle time**. Read-write. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from an MRS command to another MRS command, all to the same chip select. Bits: Description 1h-0h: Reserved Fh-2h: <Tmrd> clocks

D18F2x224_dct[0] DDR3 DRAM Timing 8

Reset: 0000_0408h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:24	**Trcpage: row cycle time per page**. Read-write. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum average time within a refresh window from an activate command to another activate command, all to the same chip select bank and row. Bits: Description 00h: Trcpage disabled. FFh-01h: <Trcpage> * 10 ns.
23:12	Reserved.
11:8	**Tzqcs: Zq short cal command delay**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from a ZQCS command to any other command (excluding NOP and DES) on the channel. Bits: Description 0h: Reserved. Fh-1h: <Tzqcs> * 16 clocks.
7:5	Reserved.
4:0	**Tzqoper: Zq long cal command delay**. Read-write. BIOS: See 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time in memory clock cycles from a ZQCL command to any other command (excluding NOP and DES) on the channel. Bits: Description 00h: Reserved. 1Fh-01h: <Tzqoper> * 32 clocks.

D18F2x228_dct[0] DDR3 DRAM Timing 9

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:24	**Tstag3: auto refresh stagger time for logical DIMM 3**. See: Tstag0.
23:16	**Tstag2: auto refresh stagger time for logical DIMM 2**. See: Tstag0.
15:8	**Tstag1: auto refresh stagger time for logical DIMM 1**. See: Tstag0.
7:0	**Tstag0: auto refresh stagger time for logical DIMM 0**. Read-write. Reset: 00h. BIOS: MAX(D18F2x204_dct[0]_mp[1:0][Trrd], CEIL(D18F2x204_dct[0]_mp[1:0][FourActWindow]/4)). Specifies the number of clocks between auto refresh commands to different ranks of a DIMM when D18F2x90_dct[0][StagRefEn] == 1. Bits: Description 00h: 0 clocks. FEh-01h: <Tstag0> clocks. FFh: 255 clocks.

D18F2x22C_dct[0]_mp[1:0] DDR3 DRAM Timing 10

Reset: 0000_000Ch. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:24	**Tvrefdq: vref dq command time**. Read-write. Specifies minimum number of memory clocks between Vref MR commands. BIOS must program such that Tvrefdq >= Tmod. Reserved if !Ddr4Mode. Bits: Description FFh-00h: <Tvrefdq> clocks.
23:13	Reserved.
12:8	**TclAdj: CAS latency adjust**. Read-write. BIOS: MAX(D18F2x200_dct[0]_mp[1:0] - 18, 0). Specifies the time in memory clock cycles from the CAS assertion for a read cycle until the signal for an outgoing read request is delivered to the phy. Bits: Description 1Fh-00h: <TclAdj> clocks.
7:5	Reserved.
4:0	**Twr: write recovery**. Read-write. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time from the last data write until the chip select bank precharge. Bits: Description 04h-00h: Reserved. 1Ah-05h: <Twr> clocks. 1Fh-1Bh: Reserved.

D18F2x[234:230]_dct[0] DDR3 DRAM Read ODT Pattern [High:Low]

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. This register is used by BIOS to specify the state of the ODT pins during DDR reads. F2x230 is used to control chip selects 0-3. F2x234 is used to control chip selects 4-7.

Table 173: Field Mapping for D18F2x[234:230]_dct[0]

Register	Bits			
	27:24	19:16	11:8	3:0
D18F2x230_dct[0]	CS3	CS2	CS1	CS0
D18F2x234_dct[0]	CS7	CS6	CS5	CS4

Bits	Description
31:28	Reserved.
27:24	**RdOdtPatCs73: read ODT pattern chip select [7,3]**. See: RdOdtPatCs40.
23:20	Reserved.
19:16	**RdOdtPatCs62: read ODT pattern chip select [6,2]**. See: RdOdtPatCs40.
15:12	Reserved.
11:8	**RdOdtPatCs51: read ODT pattern chip select [5,1]**. See: RdOdtPatCs40.
7:4	Reserved.
3:0	**RdOdtPatCs40: read ODT pattern chip select [4,0]**. Read-write. Specifies the state of ODT[3:0] pins when a read occurs to the specified chip select.

D18F2x[23C:238]_dct[0] DDR3 DRAM Write ODT Pattern [High:Low]

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. This register is used by BIOS to specify the state of the ODT pins during DDR writes. F2x238 is used to control chip selects 0-3. F2x23C is used to control chip selects 4-7.

Table 174: Field Mapping for D18F2x[23C:238]_dct[0]

Register	Bits			
	27:24	19:16	11:8	3:0
D18F2x238_dct[0]	CS3	CS2	CS1	CS0
D18F2x23C_dct[0]	CS7	CS6	CS5	CS4

Bits	Description
31:28	Reserved.
27:24	**WrOdtPatCs73: write ODT pattern chip select [7,3]**. See: WrOdtPatCs40.
23:20	Reserved.
19:16	**WrOdtPatCs62: write ODT pattern chip select [6,2]**. See: WrOdtPatCs40.
15:12	Reserved.
11:8	**WrOdtPatCs51: write ODT pattern chip select [5,1]**. See: WrOdtPatCs40.
7:4	Reserved.
3:0	**WrOdtPatCs40: write ODT pattern chip select [4,0]**. Read-write. Specifies the state of ODT[3:0] pins when a write occurs to the specified chip select.

D18F2x240_dct[0]_mp[1:0] DDR3 DRAM ODT Control

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:16	Reserved.
15:12	**WrOdtOnDuration: write ODT on duration.** Read-write. BIOS: IF (Ddr4Mode & D18F2x90_dct[0][WrCrcEn]) THEN 7. ELSE 6. ENDIF. Specifies the number of memory clock cycles that ODT is asserted for writes. Bits: Description 5h-0h: Reserved. Fh-6h: <WrOdtOnDuration> clocks.
11	Reserved.
10:8	**WrOdtTrnOnDly: Write ODT Turn On Delay**. Read-write. BIOS: 0. Specifies the number of memory clock cycles that ODT assertion is delayed relative to write CAS. Bits: Description 0h: 0 clocks. 7h-1h: <WrOdtTrnOnDly> clocks, Reserved if (WrOdtOnDuration == 0).
7:4	**RdOdtOnDuration: Read ODT On Duration**. Read-write. BIOS: 6. Specifies the number of memory clock cycles that ODT is asserted for an eight-beat read burst. The controller shortens the ODT pulse duration by two clock cycles if the burst is chopped. Bits: Description 5h-0h: Reserved. Ah-6h: <RdOdtOnDuration> clocks. Fh-Bh: Reserved.
3:0	**RdOdtTrnOnDly: Read ODT Turn On Delay.** Read-write. BIOS: MAX(0, D18F2x200_dct[0]_mp[1:0][Tcl] - D18F2x20C_dct[0]_mp[1:0][Tcwl]). Specifies the number of clock cycles that ODT assertion is delayed relative to read CAS. Bits: Description 0h: 0 clocks. Fh-1h: <RdOdtTrnOnDly> clocks, Reserved if (RdOdtOnDuration == 0).

D18F2x244_dct[0] DRAM Controller Miscellaneous 3

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:8	Reserved.
7:4	Reserved.
3:0	**PrtlChPDDynDly: partial channel power down dynamic delay**. Read-write. BIOS: 4h. Specifies the channel idle hysteresis for fast exit/slow exit mode changes when D18F2xA8_dct[0][PrtlChPDEnhEn] == 1. Bits: Description 0h: 0 clocks. 7h-1h: <PrtlChPDDynDly*32> clocks. 8h: 256 clocks. Fh-9h: Reserved.

D18F2x248_dct[0]_mp[1:0] DRAM Power Management 0

Reset: 0000_0A03h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:30	Reserved.
29:24	**AggrPDDelay: aggressive power down delay**. Read-write. BIOS: 20h. Specifies a hysteresis count from the last DRAM activity for the DCT to close pages prior to precharge power down. Reserved if D18F2xA8_dct[0][AggrPDEn] == 0. See PchgPDEnDelay and D18F2x94_dct[0][PowerDownEn]. Bits: Description 00h: 64 clocks 01h: 1 clock 3Eh-02h: <AggrPDDelay> clocks 3Fh: 63 clocks
23:22	Reserved.
21:16	**PchgPDEnDelay: precharge power down entry delay**. Read-write. BIOS: 20h. Specifies the power down entry delay. If D18F2xA8_dct[0][AggrPDEn] == 0, this delay behaves as a hysteresis. This field must satisfy the minimum power down entry delay requirements. See also D18F2x94_dct[0][PowerDownEn]. Bits: Description 00h: 64 clocks 01h: 1 clock 3Eh-02h: <PchgPDEnDelay> clocks 3Fh: 63 clocks
15:13	Reserved.
12:8	**Txpdll: exit DLL and precharge powerdown to command delay**. Read-write. Specifies the minimum time that the DCT waits to issue a command after exiting precharge powerdown mode if the DLL was also disabled. Reserved if !Ddr3Mode. Bits: Description 09h-00h: Reserved. 1Fh-0Ah: <Txpdll> clocks.
7:4	Reserved.
3:0	**Txp: exit precharge PD to command delay**. Read-write. Specifies the minimum time that the DCT waits to issue a command after exiting precharge powerdown mode. Bits: Description 2h-0h: Reserved. Fh-3h: <Txp> clocks.

D18F2x24C_dct[0] DDR3 DRAM Power Management 1

Reset: 0214_0843h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:30	Reserved.

29:24	**Tcksrx: clock stable to self refresh exit delay**. Read-write. BIOS: 0. Specifies the minimum time in memory clock cycles that the DCT waits to assert CKE after clock frequency is stable. Bits Description 3Fh-00h <Tcksrx> clocks.
23:22	Reserved.
21:16	**Tcksre: self refresh to command delay**. Read-write. Specifies the minimum time in memory clock cycles that the DCT waits to remove external clocks after entering self refresh or powerdown. Bits Description 04h-00h Reserved. 3Fh-05h <Tcksre> clocks.
15:14	Reserved.
13:8	**Tckesr: self refresh to command delay**. Read-write. BIOS:2. Specifies the minimum time in memory clock cycles that the DCT waits to issue a command after entering self refresh. Bits Description 01h-00h Reserved. 3Fh-02h <Tckesr> clocks.
7:4	**Tcpded: command pass disable delay**. Read-write. BIOS: IF (Ddr3Mode && D18F2x90_dct[0][UnbuffDimm]) THEN 2. ELSE 4. ENDIF. Specifies minimum time in memory clock cycles for powerdown entry to command bus tristate. Bits Description 0h Reserved. Fh-1h <Tcpded> clocks.
3:0	**Tpd: minimum power down entry to exit**. Read-write. Specifies minimum time in memory clock cycles for powerdown entry to exit timing. Bits Description 0h Reserved. Fh-1h <Tpd> clocks.

D18F2x250_dct[0] DRAM Loopback and Training Control

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:21	Reserved.
20:18	**DataPatGenSel: data pattern generator select**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Bits Description 000b PRBS23 I. 001b PRBS23 II. Within a byte lane, the DQ[n] value is offset 8n bit times from DQ[0]. 010b Configurable data pattern. See D18F2x2[B4,B0,AC,A8]_dct[0]. 011b Configurable data pattern with circular lane shift. See D18F2x2[B4,B0,AC,A8]_dct[0]. 100b PRBS23 III. Within a byte lane, each DQ[n] value is equal to DQ[0]. 111b-101b Reserved.
17	**ActPchgGenEn: activate precharge generation enable**. IF ((D18F2x78_dct[0][DramSprLock])) THEN Read-only. ELSE Read-write. ENDIF. 1=The DCT generates ACT and PRE traffic in the available command bandwidth from SendCmd. 0=Traffic generation is not enabled. In usage, user should clear ActPchgGenEn prior to SendCmd. Reserved if ~CmdTestEnable. Reserved if ~Ddr3Mode.

16:14	Reserved.
13	**LfsrRollOver: LFSR roll over**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Specifies the behavior of DataPrbsSeed and the data comparison logic if the generated address wraps around to equal D18F2x25[8,4]_dct[0][TgtAddress]. 0=The PRBS is not re-seeded. 1=The PRBS is re-seeded.
12	**CmdSendInProg: command in progress**. Read-only; Updated-by-hardware. 0=DCT is idle. 1=DCT is busy.
11	**SendCmd: send command**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. 0=Stop command generation. 1=Begin command generation as specified in CmdTgt, CmdType, and D18F2x260_dct[0][CmdCount]. BIOS must set this field to a 0 after a command series is completed. Reserved if ~CmdTestEnable.
10	**TestStatus: test status**. Read-only. 0=Command generation is in progress. 1=Command generation has completed. Reserved if ~(SendCmd & (D18F2x260_dct[0][CmdCount] > 0 \| StopOnErr)).
9:8	**CmdTgt: command target**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Specifies the SendCmd command target address mode. See D18F2x25[8,4]_dct[0]. Bits — Description 00b — Issue commands to address Target A 01b — Issue alternating commands to address Target A and Target B 11b-10b — Reserved.
7:5	**CmdType: command type**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Specifies the SendCmd command type. Bits — Description 000b — Read 001b — Write 010b — Alternating write and read 111b-011b — Reserved.
4	**StopOnErr: stop on error**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. Specifies the DCT behavior if a data comparison error occurs. 1=Stop command generation. 0=Continue command generation. If StopOnErr == 1, BIOS must program ResetAllErr = 1 when programming SendCmd = 1.
3	**ResetAllErr: reset all errors**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read; Write-1-only; Cleared-by-hardware. ENDIF. 1=Clear error status bits and error counters in D18F2x264_dct[0], D18F2x268_dct[0], and D18F2x26C_dct[0].
2	**CmdTestEnable: command test enable**. IF (D18F2x78_dct[0][DramSprLock]) THEN Read-only. ELSE Read-write. ENDIF. 0=Disable the command generation mode. 1=Enable the command generation mode. See SendCmd. Software must disable data scrambling before using this logic to generate patterns. See D18F2x90_dct[0][DataScrambleEn].
1:0	Reserved.

D18F2x25[8,4]_dct[0] DRAM Target [B, A] Base

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Table 175: Register Mapping for D18F2x25[8,4]_dct[0]

Register	Function
D18F2x254_dct[0]	Target A
D18F2x258_dct[0]	Target B

Bits	Description
31:27	Reserved.
26:24	**TgtChipSelect: target chip select**. Read-write. Specifies the chip select. Bits: Description 111b-000b: CS<TgtChipSelect>
23:20	**TgtBank: target bank [3:0]**. Read-write. Specifies the bank address. If Ddr3Mode then TgtBank[3] is ignored.
19:16	Reserved.
15:10	**TgtAddress[15:10]: target address [15:10]**. Read-write. Specifies the upper column address bits[15:10]. Software must always program bit[10] and bit[12] equal to 0.
9:0	**TgtAddress[9:0]: target address [9:0]**. Read-write. Specifies the column address bits[9:0]. The address sequencing in a command series occurs as follows: TgtAddress[9:3] is incremented by one with wrap around. The increment occurs after each command if D18F2x250_dct[0][CmdType] == 00Xb or if (D18F2x250_dct[0][CmdType] == 010b and D18F2x250_dct[0][CmdTgt] == 01b). The increment occurs after each command pair if (D18F2x250_dct[0][CmdType] == 010b and D18F2x250_dct[0][CmdTgt] == 00b).

D18F2x25C_dct[0] DRAM Command 0

Reset: 0000_0001h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:22	**BubbleCnt2: bubble count 2**. See: BubbleCnt. Specifies the number of NOP commands inserted after the last clock of virtual CAS of each read-burst operation in alternating write and read mode. Defined only if (D18F2x250_dct[0][CmdType] == 010b); otherwise reserved.
21:12	**BubbleCnt: bubble count**. Read-write. Specifies the number of NOP commands inserted after the last clock of virtual CAS of the last command of the command stream specified by CmdStreamLen. Bits: Description 000h: 0 command bubbles 3FEh-001h: <BubbleCnt> command bubbles 3FFh: 3FFh command bubbles
11:9	Reserved.

8	Reserved.
7:0	**CmdStreamLen: command stream length**. Read-write. Specifies the number of commands generated before BubbleCnt bubbles are inserted. Bits: Description 00h: Reserved 01h: 1 command FEh-02h: <CmdStreamLen> commands; defined only if ~(D18F2x250_dct[0][CmdType]==010b) FFh: 255 commands; defined only if ~(D18F2x250_dct[0][CmdType]==010b)

D18F2x260_dct[0] DRAM Command 1

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:21	Reserved.
20:0	**CmdCount: command count**. Read-write. Specifies the maximum number of commands to generate when D18F2x250_dct[0][SendCmd] == 1. See also D18F2x250_dct[0][StopOnErr]. Bits: Description 00_0000h: Infinite commands. 1F_FFFFh-00_0001h: <CmdCount> commands.

D18F2x264_dct[0] DRAM Status 0

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:25	**ErrDqNum: error DQ number**. Read-only. Indicates the DQ bit of the first error occurrence when D18F2x264_dct[0][ErrCnt] > 0. Cleared by D18F2x250_dct[0][ResetAllErr]. Bits: Description 3Fh-00h: Data[<ErrDqNum>] 7Fh-40h: Reserved.
24:0	**ErrCnt: error count**. Read; Set-by-hardware; Write-1-to-clear. Specifies a saturating counter indicating the number of DQ bit errors detected. Counts a maximum of 1 error per bit-lane per each bit-time. Status is accumulated until cleared by D18F2x250_dct[0][ResetAllErr]. Errors can be masked on per-bit basis by programming D18F2x274_dct[0] and D18F2x278_dct[0]. Bits: Description 000_0000h: 0 errors 1FF_FFFDh-000_0001h: <ErrCnt> errors 1FF_FFFEh: 1FF_FFFEh errors 1FF_FFFFh: 1FF_FFFFh or more errors

D18F2x268_dct[0] DRAM Status 1

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:20	Reserved.
19:0	**NibbleErrSts: nibble error status**. Read-only. Indicates error detection status on a per nibble basis when D18F2x264_dct[0][ErrCnt] > 0. Status is accumulated until cleared by D18F2x250_dct[0][ResetAllErr]. Bit Description [15:0] Data[<(NibbleErrSts*4)+3>:<NibbleErrSts*4>] [19:16] Reserved

D18F2x26C_dct[0] DRAM Status 2

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:18	Reserved.
17:0	**NibbleErr180Sts: nibble error 180 status**. Read-only. Indicates error detection status on a per nibble basis when D18F2x264_dct[0][ErrCnt] > 0, comparing read data against data shifted 1-bit time earlier. Status is accumulated until cleared by D18F2x250_dct[0][ResetAllErr]. Bit Description [15:0] Data[<(NibbleErr180Sts]*4)+3>:<NibbleErr180Sts*4>] [19:16] Reserved

D18F2x270_dct[0] DRAM PRBS

See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:19	Reserved.
18:0	**DataPrbsSeed: data PRBS seed**. Read-write. Reset: 7FFFFh. Specifies the seed value used for creating pseudo random traffic on the data bus. This register must be written with a non-zero seed value.

D18F2x274_dct[0] DRAM DQ Mask Low

See D18F1x10C[DctCfgSel]. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:0	**DQMask[31:0]: DQ mask**. Read-write. DQMask[63:0] = {D18F2x278_dct[0][DQMask[63:32]], DQMask[31:0]}. Reset: 0000_0000_0000_0000h. 1=The corresponding DQ bit is not compared. 0=The corresponding DQ bit is compared. See D18F2x264_dct[0][ErrCnt]. Bit Description [0] Data[0] [62:1] Data[<DQMask>] [63] Data[63]

D18F2x278_dct[0] DRAM DQ Mask High

Bits	Description
31:0	**DQMask[63:32]: DQ mask**. See: D18F2x274_dct[0][DQMask[31:0]].

D18F2x280_dct[0] DRAM DQ Pattern Override 0

See D18F1x10C[DctCfgSel]. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:0	**DQPatOvrEn[31:0]: DQ pattern override enable**. Read-write. DQPatOvrEn[63:0] = {D18F2x284_dct[0][DQPatOvrEn[63:32]], DQPatOvrEn[31:0]}. Reset: 0000_0000_0000_0000h. DQ pattern field depends on usage scenario. 1=IF D18F2x7C_dct[0][SendPDAMrsCmd] THEN This DQ bit lane is overridden with a static '1' value. ELSE The pattern on this DQ bit lane is overridden with a static value specified by D18F2x288_dct[0][PatOvrVal]. ENDIF. 0=IF D18F2x7C_dct[0][SendPDAMrsCmd] THEN This DQ bit lane is overridden with a static '0' value. ELSE The pattern on this DQ bit lane is not verridden. ENDIF. Reserved if CmdTestEnable && D18F2x250_dct[0][DataPatGenSel] != 10b or D18F2x7C_dct[0][SendPDAMrsCmd] != 1. Bit Description [0] Data[0] [62:1] Data[<DQPatOvrEn[31:0]>] [63] Data[63]

D18F2x284_dct[0] DRAM DQ Pattern Override 1

Bits	Description
31:0	**DQPatOvrEn[63:32]: DQ pattern override enable**. See: D18F2x280_dct[0][DQPatOvrEn[31:0]].

D18F2x288_dct[0] DRAM DQ Pattern Override 2

Reset: 0000_0000h. See D18F1x10C[DctCfgSel]. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:24	**XorPatOvr: xor pattern override**. Read-write. Specifies the override data pattern used for creating traffic on the data bus. The output data for each DQ of each byte lane is XOR'd with the specified values. Reserved if D18F2x250_dct[0][DataPatGenSel] != 100b. Bit — Description [0] — DQ[0] [1] — DQ[1] [6:2] — DQ[<XorPatOvr>] [7] — DQ[7]
23:9	Reserved.
8	**PatOvrVal: pattern override value**. Read-write. Specifies the override data pattern used for creating traffic on the data bus. 1=Static 1's. 0=Static 0's. Reserved if D18F2x250_dct[0][DataPatGenSel] != 10b.
7:0	Reserved.

D18F2x28C_dct[0] DRAM Command 2

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation]. This register may only be used when D18F2x250_dct[0][CmdTestEnable] == 1.

Bits	Description
31	**SendActCmd: send activate command**. Read; Write-1-only; Cleared-by-hardware. 1=The DCT sends an activate command as specified by ChipSelect, Bank, and Address. This bit is cleared by hardware after the command completes.
30	**SendPchgCmd: send precharge all command**. Read; Write-1-only; Cleared-by-hardware. The DCT sends a precharge command based on CmdAddress[10]. This bit is cleared by hardware after the command completes. 0=Command has completed. 1=If (CmdAddress[10] == 1) then send a precharge all command as specified by CmdChipSelect; If (CmdAddress[10] == 0) then send a precharge command as specified by CmdChipSelect, CmdBank.
29:22	**CmdChipSelect: command chip select**. Read-write. Specifies the chip select. Bit — Description [7:0] — CS<CmdChipSelect>
21:18	**CmdBank: command bank [3:0]**. Read-write. Specifies the bank address. Bits — Description 7h-0h — Bank<CmdBank> Fh-8h — Reserved.
17:0	**CmdAddress: command address [17:0]**. Read-write. Specifies the row address.

D18F2x290_dct[0] DRAM Status 3

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:27	Reserved.
26:24	**ErrBeatNum: error beat number**. Read-only. Indicates the data beat of the first error occurrence in the command reported by ErrCmdNum when D18F2x264_dct[0][ErrCnt] > 0 and D18F2x260_dct[0][CmdCount] > 0. Cleared by D18F2x250_dct[0][ResetAllErr]. Bits: Description 7h-0h: <ErrBeatNum> beat
23:21	Reserved.
20:0	**ErrCmdNum: error command number**. Read-only. Indicates the command number of the first error occurrence when D18F2x264_dct[0][ErrCnt] > 0 and D18F2x260_dct[0][CmdCount] > 0. Cleared by D18F2x250_dct[0][ResetAllErr]. Bits: Description 1F_FFFFh-00_0000h: <ErrCmdNum> command

D18F2x294_dct[0] DRAM Status 4

See 2.9.3 [DCT Configuration Registers]. See 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:0	**DQErr[31:0]: DQ error**. Read-only. DQErr[63:0] = {D18F2x298_dct[0][DQErr[63:32]], DQErr[31:0]}. Reset: 0000_0000_0000_0000h. Indicates error detection status on a per bit basis when D18F2x264_dct[0][ErrCnt] > 0. Status is accumulated until cleared by D18F2x250_dct[0][ResetAllErr]. Bit: Description [0]: Data[0] [62:1]: Data[<DQErr>] [63]: Data[63]

D18F2x298_dct[0] DRAM Status 5

Bits	Description
31:0	**DQErr[63:32]: DQ error**. See: D18F2x294_dct[0][DQErr[31:0]].

D18F2x2A0_dct[0]_mp[1:0] DRAM Timing 12

Bits	Description
31:27	Reserved.
26:16	**Trfc1: auto refresh row cycle time DIMM1**. See: D18F2x2A0_dct[0]_mp[1:0][Trfc0].

15:11	Reserved.
10:0	**Trfc0: auto refresh row cycle time DIMM0**. Read-write. Reset: 0000_0420h. BIOS: 2.9.9.3 [SPD ROM-Based Configuration]. Specifies the minimum time from a refresh command to the next valid command, except NOP or DES. The recommended programming of this register varies based on DRAM density and speed. Bits Description 000h Reserved 7FFh-001h <Trfc0> clocks

D18F2x2A4_dct[0]_mp[1:0] DRAM Timing 13

Bits	Description
31:27	Reserved.
26:16	**Trfc3: auto refresh row cycle time DIMM1**. See: D18F2x2A0_dct[0]_mp[1:0][Trfc0].
15:11	Reserved.
10:0	**Trfc2: auto refresh row cycle time DIMM0**. See: D18F2x2A0_dct[0]_mp[1:0][Trfc0].

D18F2x2[B4,B0,AC,A8]_dct[0] DRAM User Data Pattern

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers] and 2.9.10 [Continuous Pattern Generation].

Table 176: Register Mapping for D18F2x2[B4,B0,AC,A8]_dct[0]

Register	Function
D18F2x2A8_dct[0]	Nibble Lanes 0, 4, 8, 12, 16
D18F2x2AC_dct[0]	Nibble Lanes 1, 5, 9, 13, 17
D18F2x2B0_dct[0]	Nibble Lanes 2, 6, 10, 14
D18F2x2B4_dct[0]	Nibble Lanes 3, 7, 11, 15

Table 177: Field Mapping for D18F2x2[B4,B0,AC,A8]_dct[0]

Register	Bits			
	31:24	23:16	15:8	7:0
D18F2x2A8_dct[0]	DQ3_DQ19_DQ35_DQ51	DQ2_DQ18_DQ34_DQ50	DQ1_DQ17_DQ33_DQ49	DQ0_DQ16_DQ32_DQ48
D18F2x2AC_dct[0]	DQ7_DQ23_DQ39_DQ55	DQ6_DQ22_DQ38_DQ54	DQ5_DQ21_DQ37_DQ53	DQ4_DQ20_DQ36_DQ52
D18F2x2B0_dct[0]	DQ11_DQ27_DQ43_DQ59	DQ10_DQ26_DQ42_DQ58	DQ9_DQ25_DQ41_DQ57	DQ8_DQ24_DQ40_DQ56
D18F2x2B4_dct[0]	DQ15_DQ31_DQ47_DQ63	DQ14_DQ30_DQ46_DQ62	DQ13_DQ29_DQ45_DQ61	DQ12_DQ28_DQ44_DQ60

Bits	Description
31:24	**DataPattern: data pattern**. See: D18F2x2[B4,B0,AC,A8]_dct[0][7:0].
23:16	**DataPattern: data pattern**. See: D18F2x2[B4,B0,AC,A8]_dct[0][7:0].

15:8	**DataPattern: data pattern**. See: D18F2x2[B4,B0,AC,A8]_dct[0][7:0].
7:0	**DataPattern: data pattern**. Read-write. Specifies a data pattern used for creating traffic on the data bus for the specified bit lanes. See D18F2x250_dct[0][DataPatGenSel]. Bit — Description [0] — Pattern Data Beat 0 [6:1] — Pattern Data Beat 1 to Pattern Data Beat 6 [7] — Pattern Data Beat 7

D18F2x2B8_dct[0] DRAM Command 3

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers] and 2.9.10 [Continuous Pattern Generation].

Bits	Description
31:16	**ActPchgSeq: activate precharge sequence**. Read-write. Specifies the command that is generated in the sequence, corresponding to the sixteen target entries in D18F2x2[C0,BC]_dct[0]. 1=Activate. 0=Precharge. Bit — Description [0] — ACT/PRE [0] [14:1] — ACT/PRE [<ActPchgSeq>] [15] — ACT/PRE [15]
15:14	Reserved.
13:8	**ActPchgCmdMin: activate precharge command minimum**. Read-write. Specifies the minimum time in memory clock cycles from an activate or precharge command to the next activate or precharge command in the ActPchgSeq sequence. Bits — Description 00h — Reserved 3Fh-01h — [<ActPchgCmdMin>] clocks
7:0	Reserved.

D18F2x2[C0,BC]_dct[0] DRAM Command 4 & 5

Reset: 0000_0000h. See 2.9.3 [DCT Configuration Registers] and 2.9.10 [Continuous Pattern Generation].

Table 178: Field Mappings for D18F2x2[C0,BC]_dct[0]

Register	Bits							
	31:28	27:24	23:20	19:16	15:12	11:8	7:4	3:0
D18F2x2BC_dct[0]	ACT_PRE 7	ACT_PRE 6	ACT_PRE 5	ACT_PRE 4	ACT_PRE 3	ACT_PRE 2	ACT_PRE 1	ACT_PRE 0
D18F2x2C0_dct[0]	ACT_PRE 15	ACT_PRE 14	ACT_PRE 13	ACT_PRE 12	ACT_PRE 11	ACT_PRE 10	ACT_PRE 9	ACT_PRE 8

Bits	Description
31:28	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].
27:24	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].
23:20	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].
19:16	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].

15:12	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].
11:8	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].
7:4	**ActPchgTgtBank: activate precharge target bank**. See: D18F2x2[C0,BC]_dct[0][3:0].
3:0	**ActPchgTgtBank: activate precharge target bank**. Read-write. Specifies the bank address used for this command of the sequence. Bits Description Fh-0h Bank [<ActPchgTgtBank>] ActPchgTgtBank[3] is reserved if !Ddr4Mode.

D18F2x2E0_dct[0] Memory P-state Control and Status

See 2.9.3 [DCT Configuration Registers].

Bits	Description
31	Reserved.
30	**FastMstateDis: fast M-state change disable**. Read-write. Reset: 0. 1=The DCT changes MEMCLK frequency only after the NCLK frequency has changed. 0=The DCT changes MEMCLK frequency while the Northbridge changes NCLK.
29	Reserved.
28:24	**M1MemClkFreq: M1 memory clock frequency**. Read-write. Reset: 00h. Specifies the frequency of the DRAM interface (MEMCLK) for memory P-state 1. Table 118 [Valid Values for Memory Clock Frequency Value Definition]. The hardware enforces D18F5x84[DdrMaxRateEnf] when writes to this field occur. See D18F5x84[DdrMaxRate] and D18F5x84[DdrMaxRateEnf].
23:16	**MxMrsEn: Mx Mrs enable**. Read-write. Reset: 00h. 1=The DCT writes to the DRAM MR the values in D18F2x2[F4:E8]_dct[0]_mp[1:0] after a memory P-state change or D18F2x90_dct[0][ExitSelfRef]. 0=The DCT does not write to the DRAM MR. BIOS should should only enable required MRS commands to optimize P-state switching latency. Bit Description, MR value [0] MR0 [1] MR1 [2] MR2 [3] MR3 [4] MR4 [5] MR5 [6] MR6 [7] Reserved
15:6	Reserved.
5	**VrefTrainMrsEn: Vref training MRS enable**. Read-write. Reset: 0. 1=The DCT writes to clear the DRAM MR6 training enable Tvrefdq clocks after a memory P-state change. 0=The DCT does not write to the DRAM MR. Reserved if !Ddr4Mode. Reserved if !MxMrsEn[6].
4	**VrefTrain3Mrs6: Vref training 3 MRS enable**. Read-write. Reset: 0. 1=The DCT updates the DRAM MR6 using 3 writes. 0=The DCT does not write to the DRAM MR. Reserved if !(Ddr4Mode & Ddr4VrefTrainMrsEn).

3:1	Reserved.
0	**CurMemPstate: current memory P-state**. Read-only; Updated-by-hardware. Reset: 0. Specifies the current memory P-state. 0=M0. 1=M1.

D18F2x2[F4:E8]_dct[0]_mp[1:0] MRS Buffer

See 2.9.3 [DCT Configuration Registers].

Table 179: Field Mappings for D18F2x2[F4:E8]_dct[0]_mp[1:0]

Register	Bits	
	31:16	15:0
D18F2x2E8_dct[0]_mp[1:0]	MR1	MR0
D18F2x2EC_dct[0]_mp[1:0]	MR3	MR2
D18F2x2F0_dct[0]_mp[1:0]	MR5	MR4
D18F2x2F4_dct[0]_mp[1:0]	Reserved	MR6

Bits	Description
31:16	**MxMr: Mx MR**. See: D18F2x2[F4:E8]_dct[0]_mp[1:0][15:0].
15:0	**MxMr: Mx MR**. Read-write. Reset: 0000h. Specifies the value written by hardware to the DRAM MR if enabled in D18F2x2E0_dct[0][MxMrsEn]. Corresponds to A[15:0] in Ddr3Mode. Corresponds to {A[17], 0, A[13:0]} in Ddr4Mode.

D18F2x2FC_dct[0] DRAM Timing 11

Reset: 0000_0004h. See 2.9.3 [DCT Configuration Registers].

Bits	Description
31:0	Reserved.

3.13 Device 18h Function 3 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D18F3x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 15B3h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D18F3x04 Status/Command

Bits	Description
31:16	**Status**. Read-only. Reset: 0000h, except bit[20]. Bit[20] is set to indicate the existence of a PCI-defined capability block, if one exists.
15:0	**Command**. Read-only. Reset: 0000h.

D18F3x08 Class Code/Revision ID

Bits	Description
31:8	**ClassCode**. Read-only. Reset: 06_0000h. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only. Reset: 00h.

D18F3x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:0	**HeaderTypeReg**. Read-only. These bits are fixed at their default values. The header type field indicates that there are multiple functions present in this device.

D18F3x34 Capability Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr**. Read-only. Value: F0h. Points to D18F3xF0 to provide capability.

D18F3x40 MCA NB Control

MSR0000_0410[31:0] is an alias of D18F3x40. See MSR0000_0410[31:0].

MSR0000_0410[31:0] is an alias of D18F3x40, which is accessible through PCI configuration space. Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn].

See D18F3x44 [MCA NB Configuration] for further NB MCA configuration controls. See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture.
See MSRC001_0048 [NB Machine Check Control Mask (MC4_CTL_MASK)] for the corresponding error mask register.
See Table 182 [MC4 Error Descriptions] and Table 183 [MC4 Error Signatures, Part 1].

Bits	Description
31	**McaCpuDatErrEn: Compute Unit data error**. Read-write. Reset: 0. 1=Enables MCA reporting of CPU data errors sent to the NB.
30	Unused.
29	**McaWrCrcErrEn**. Read-write. Reset: 0. 1=DRAM write CRC error logging enable.
28	**G5CrcCorrErren**. Read-write. Reset: 0. 1=DRAM CRC correctable error enable.
27	Reserved.
26	**NbArrayParEn: northbridge array parity error reporting enable**. Read-write. Reset: 0. 1=Enables reporting of parity errors in the NB arrays.
25	**UsPwDatErrEn: upstream data error enable**. Read-write. Reset: 0. 1=Enables MCA reporting of upstream posted writes in which the EP bit is set.
24	**SyncPktEn[3]: link sync packet error reporting enable for link 3**. Read-write. Reset: 0. See: SyncPktEn[2:0].
23	**CrcErrEn[3]: link CRC error reporting enable for link 3**. Read-write. Reset: 0. See: CrcErrEn[2:0].
22:19	**RtryHtEn: link retry reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of retries on the link. Bit Description [3] Link 3 [2] Link 2 [1] Link 1 [0] Link 0
18	**DramParEn: DRAM parity error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of parity errors on the DRAM address or control signals.
17	**CpPktDatEn: completion packet error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of completion packets with the EP bit set.
16	**NbIntProtEn: northbridge internal bus protocol error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of protocol errors detected on the northbridge internal bus. When possible, this enable should be cleared before initiating a warm reset to avoid logging spurious errors due to RESET_L signal skew.

15	**NbAryUncEn**. Read-write. Reset: 0. 1=Enables MCA reporting of uncorrectable errors in the North-bridge arrays.
14	**NbAryCorEn**. Read-write. Reset: 0. 1=Enables MCA reporting of correctable errors in the North-bridge arrays.
13	**DevErrEn**. Read-write. Reset: 0.
12	**WDTRptEn: watchdog timer error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of watchdog timer errors. The watchdog timer checks for NB system accesses for which a response is expected but no response is received. See D18F3x44 [MCA NB Configuration] for information regarding configuration of the watchdog timer duration. This bit does not affect operation of the watchdog timer in terms of its ability to complete an access that would otherwise cause a system hang. This bit only affects whether such errors are reported through MCA.
11	**AtomicRMWEn: atomic read-modify-write error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of atomic read-modify-write (RMW) commands received from an IO link. Atomic RMW commands are not supported. An atomic RMW command results in a link error response being generated back to the requesting IO device. The generation of the link error response is not affected by this bit.
10	Reserved.
9	**TgtAbortEn: target abort error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of target aborts to a link. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of this bit.
8	**MstrAbortEn: master abort error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of master aborts to a link. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of this bit.
7:5	**SyncPktEn[2:0]: link sync packet error reporting enable for links [2:0]**. Read-write. Reset: 0. SyncPktEn[3:0] = {SyncPktEn[3], SyncPktEn[2:0]}. 1=Enables MCA reporting of link-defined sync error packets detected on link. The NB floods its outgoing links with sync packets after detecting a sync packet on an incoming link independent of the state of this bit. Bit Description [3] Link 3 [2] Link 2 [1] Link 1 [0] Link 0
4:2	**CrcErrEn[2:0]: link CRC error reporting enable for links [2:0]**. Read-write. Reset: 0. CrcErrEn[3:0] = {CrcErrEn[3], CrcErrEn[2:0]}. 1=Enables MCA reporting of CRC errors detected on link (see the description of CRC Error in Table 182 and Table 183). The NB floods its outgoing links with sync packets after detecting a CRC error on an incoming link independent of the state of this bit. Bit Description [3] Link 3 [2] Link 2 [1] Link 1 [0] Link 0
1	**UECCEn: uncorrectable ECC error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of DDR3 DRAM uncorrectable ECC errors which are detected in the NB. In some cases data may be forwarded to the core prior to checking ECC in which case the check takes place in one of the other error reporting banks.
0	**CECCEn: correctable ECC error reporting enable**. Read-write. Reset: 0. 1=Enables MCA reporting of DDR3 DRAM correctable ECC errors which are detected in the NB.

D18F3x44 MCA NB Configuration

See D18F3x180 [Extended NB MCA Configuration].
See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture.

Bits	Description
31	**NbMcaLogEn: northbridge MCA log enable**. Read-write. Reset: 0. 1=Enables logging (but not reporting) of NB MCA errors even if MCA is not globally enabled.
30	**SyncFloodOnDramAdrParErr: sync flood on DRAM address parity error**. Read-write. Reset: 0. BIOS: 1. 1=Enable Sync flood on detection of a DRAM address parity error.
29	**DisMstAbortCpuErrRsp: master abort CPU error response disable**. Read-write. Reset: 0. 1=Disables master abort reporting through the CPU MCA error-reporting banks; Suppresses sending of RDE to CPU; Does not log any MCA information in the NB.
28	**DisTgtAbortCpuErrRsp: target abort CPU error response disable**. Read-write. Reset: 0. 1=Disables target abort reporting through the CPU MCA error-reporting banks; Suppresses sending of RDE to CPU; Does not log any MCA information in the NB.
27	**NbMcaToMstCpuEn: machine check errors to master CPU only**. Read-write. Reset: 0. BIOS: 1. 1=NB MCA errors in CMP device are only reported to the node base core (NBC), and the NB MCA registers in MSR space (MSR0000_0410, MSR0000_0411, MSR0000_0412, MSR0000_0413, MSRC000_0408, and MSRC001_0048) are only accessible from the NBC; reads of these MSRs from other cores return 0's and writes are ignored. This allows machine check handlers running on different cores to avoid coordinating accesses to the NB MCA registers. This field does not affect PCI-defined configuration space accesses to these registers, which are accessible from all cores. See 3.1 [Register Descriptions and Mnemonics] for a description of MSR space and 3 [Registers] for PCI-defined configuration space. 0=NB MCA errors may be reported to the core that originated the request, if applicable and known, and the NB MCA registers in MSR space are accessible from any core. Note: • When the CPU which originated the request is known, it is stored in MSR0000_0411[ErrCoreId], regardless of the setting of NbMcaToMstCpuEn. See Table 184 for errors where ErrCoreId is known. • If IO originated the request, then the error is reported to the NBC, regardless of the setting of NbMcaToMstCpuEn.
26	**FlagMcaCorrErr: correctable error MCA exception enable**. Read-write. Reset: 0. 1=Raise a machine check exception for correctable and deferred machine check errors which are enabled in D18F3x40.
25	**DisPciCfgCpuErrRsp: PCI configuration CPU error response disable**. Read-write. Reset: 0. 1=Disables generation of an error response to the core on detection of a master abort, target abort, or data error condition, and disables logging and reporting through the MCA error-reporting banks for PCI configuration accesses. For NB WDT errors on PCI configuration accesses, this prevents sending an error response to the core, but does not affect logging and reporting of the NB WDT error. See D18F3x180[DisPciCfgCpuMstAbortRsp], which applies only to master aborts.
24	**IoRdDatErrEn: IO read data error log enable**. Read-write. Reset: 0. 1=Enables MCA logging and reporting of errors on transactions from IO devices upon detection of a target abort, master abort, or data error condition. 0=Errors on transactions from IO devices are not logged in MCA, although error responses to the requesting IO device may still be generated.
23:22	Reserved.

21	**SyncFloodOnAnyUcErr: sync flood on any UC error**. Read-write. Reset: 0. BIOS: 1. 1=Enable Sync flood of all links with sync packets on detection of any NB MCA error that is uncorrectable, including northbridge array errors and link protocol errors.
20	**SyncFloodOnWDT: sync flood on watchdog timer error**. Read-write. Reset: 0. BIOS: 1. 1=Enable Sync flood of all links with sync packets on detection of a watchdog timer error.
19:18	**GenSubLinkSel: sublink select for CRC error generation**. Read-write. Reset: 0. Selects the sublink of a link selected by GenLinkSel to be used for CRC error injection through GenCrcErrByte0 and GenCrcErrByte1. When the link is ganged, GenSubLinkSel must be 00b. When the link is unganged, the following values indicate which sublink is selected: Bits — Description 00b — Sublink 0 01b — Sublink 1 10b — Reserved 11b — Reserved
17	**GenCrcErrByte1: generate CRC error on byte lane 1**. Read-write. Reset: 0. 1=For ganged links (see GenSubLinkSel), a CRC error is injected on byte lane 1 of the link specified by GenLinkSel. For ganged links in retry mode or unganged links, this field is reserved, and GenCrcErrByte0 must be used. The data carried by the link is unaffected. This bit is cleared after the error has been generated.
16	**GenCrcErrByte0: generate CRC error on byte lane 0**. Read-write. Reset: 0. 1=Causes a CRC error to be injected on byte lane 0 of the link specified by GenLinkSel and the sublink specified by GenSubLinkSel. The data carried by the link is unaffected. This bit is cleared after the error has been generated.
15:14	**GenLinkSel: link select for CRC error generation**. Read-write. Reset: 00b. Selects the link to be used for CRC error injection through GenCrcErrByte1/GenCrcErrByte0. Bits — Description 00b — link 0 01b — link 1 10b — link 2 11b — link 3
13:12	**WDTBaseSel: watchdog timer time base select**. Read-write. Reset: 0. Selects the time base used by the watchdog timer. The counter selected by WDTCntSel determines the maximum count value in the time base selected by WDTBaseSel. Bits — Description 00b — 1.31 ms 01b — 1.28 us 10b — Reserved. 11b — Reserved.

11:9	**WDTCntSel[2:0]: watchdog timer count select bits[2:0]**. Read-write. Reset: 0. BIOS: 0010b. Selects the count used by the watchdog timer. WDTCntSel = {D18F3x180[WDTCntSel[3]], D18F3x44[WDTCntSel[2:0]]}. The counter selected by WDTCntSel determines the maximum count value in the time base selected by WDTBaseSel. WDTCntSel is encoded as: Bits — Description 0000b — 4095 0001b — 2047 0010b — 1023 0011b — 511 0100b — 255 0101b — 127 0110b — 63 0111b — 31 1000b — 8191 1001b — 16383 1111b-1010b — Reserved Because WDTCntSel is split between two registers, care must be taken when programming WDTCntSel to ensure that a reserved value is never used by the watchdog timer or undefined behavior could result.
8	**WDTDis: watchdog timer disable**. Read-write. Cold reset: 0. 1=Disables the watchdog timer. The watchdog timer is enabled by default and checks for NB system accesses for which a response is expected and where no response is received. If such a condition is detected the outstanding access is completed by generating an error response back to the requestor. An MCA error may also be generated if enabled in D18F3x40 [MCA NB Control].
7	**IoErrDis: IO error response disable**. Read-write. Reset: 0. 1=Disables setting either Error bit in link response packets to IO devices on detection of a target or master abort error condition.
6	**CpuErrDis: CPU error response disable**. Read-write. Reset: 0. BIOS: 1. 1=Disables generation of a read data error response to the core on detection of a target or master abort error condition.
5	**IoMstAbortDis: IO master abort error response disable**. Read-write. Reset: 0. 1=Signals target abort instead of master abort in link response packets to IO devices on detection of a master abort error condition.When IoMstAbortDis and D18F3x180[ChgMstAbortToNoErr] are both set, D18F3x180[ChgMstAbortToNoErr] takes precedence.
4	**SyncPktPropDis: sync packet propagation disable**. Read-write. Reset: 0. 1=Disables flooding of all outgoing links with sync packets when a sync packet is detected on an incoming link. Sync packets are propagated by default.
3	**SyncPktGenDis: sync packet generation disable**. Read-write. Reset: 0. 1=Disables flooding of all outgoing links with sync packets when a CRC error is detected on an incoming link. By default, sync packet generation for CRC errors is controlled through D18F0x[E4,C4,A4,84] [Link Control].
2	**SyncFloodOnDramUcEcc: sync flood on uncorrectable DRAM ECC error**. Read-write. Reset: 0. BIOS: 1. 1=Enable Sync flood of all links with sync packets on detection of an uncorrectable DRAM ECC error.
1	**CpuRdDatErrEn: CPU read data error log enable**. Read-write. Reset: 0. 1=Enables reporting of read data errors (master aborts and target aborts) for data destined for the CPU on this node. This bit should be clear if read data error logging is enabled for the remaining error reporting blocks in the CPU. Logging the same error in more than one block may cause a single error event to be treated as a multiple error event and cause the CPU to enter shutdown.
0	Reserved.

D18F3x48 MCA NB Status Low

See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture. MSR0000_0411[31:0] is an alias of D18F3x48. See MSRC001_0015[McStatusWrEn] for information on writing to this register.
Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. See 2.16.1 [Machine Check Architecture] for machine check architecture background.

Table 182 describes each error type. Table 183 and Table 184 describe the error codes and status register settings for each error type.

Bits	Description
31:22	Reserved.
21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware; Not-same-for-all. Cold reset: 0. Logs an extended error code when an error is detected. This model-specific field is used in conjunction with ErrorCode to identify the error sub-type for root cause analysis (see 2.16.1.5 [Error Code]). See Table 181 [MC0 Error Signatures] for expected values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware; Not-same-for-all. Cold reset: 0. See 2.16.1.5 [Error Code] for details on decoding this field. See Table 181 [MC0 Error Signatures] for expected values.

D18F3x4C MCA NB Status High

Not-same-for-all. Cold reset: 0000_0000h.
See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture. MSR0000_0411[63:32] is an alias of D18F3x4C. See MSRC001_0015[McStatusWrEn] for information on writing to this register.
Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. See 2.16.1 [Machine Check Architecture] for machine check architecture background.

Table 182 describes each error type. Table 183 and Table 184 describe the error codes and status register settings for each error type.

Bits	Description
31	**Val: error valid**. Read-write; Set-by-hardware. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.

30	**Overflow: error overflow**. Read-write; Set-by-hardware. 1=An error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten. The following hierarchy identifies the error logging priorities. 1. Uncorrectable errors 2. Deferred errors 3. Correctable errors The machine check mechanism handles the contents of MC*i*_STATUS during overflow as follows: • Higher priority errors overwrite lower priority errors. • New errors of equal or lower priority do not overwrite existing errors. • Uncorrectable errors which are not logged due to overflow result in setting PCC, unless the new uncorrectable error is of the same type and in the same reportable address range as the existing error.
29	**UC: error uncorrected**. Read-write; Updated-by-hardware. 1=The error was not corrected by hardware.
28	**En: error enable**. Read-write; Updated-by-hardware. 1=MCA error reporting is enabled for this error, as indicated by MCi_CTL.
27	**MiscV: miscellaneous error register valid**. Read-write; Updated-by-hardware. 1=Valid thresholding in MSR0000_0413 or MSRC000_0408.
26	**AddrV: error address valid**. Read-write; Updated-by-hardware. 1=MCi_ADDR contains address information associated with the error.
25	**PCC: processor context corrupt**. Read-write; Updated-by-hardware. 1=Hardware context of the processor may have been corrupted. Continued operation of the system may have unpredictable results. The error is not recoverable or survivable, and the system should be reinitialized. See 2.16.1.6.1 [Differentiation Between System-Fatal and Process-Fatal Errors].
24	**ErrCoreIdVal: error core ID is valid**. Read-write; Set-by-hardware. 1=The ErrCoreId field is valid.
23:13	Reserved.
12	**Deferred: deferred error**. Read-write; Updated-by-hardware. 1=A deferred error was created. A deferred error is the result of an uncorrectable data error which did not immediately cause a processor exception; the data is poisoned and an exception is deferred until the data is consumed.
11:10	Reserved.
9	**SubLink: sublink**. Read-write; Set-by-hardware. For errors associated with a link, this bit indicates if the error was associated with the upper or lower byte of the link. 0=Sublink [7:0]. 1=Sublink [15:8].
8:4	Reserved.
3:0	**ErrCoreId: error associated with core N**. Read-write; Updated-by-hardware. When ErrCoreIdVal == 1 this field indicates which core within the processor is associated with the error; Otherwise, this field is reserved. All values greater than D18F5x84[CmpCap] are reserved.

Table 180: MC0 Error Descriptions

Error Type	Error Sub-type	Description[1]	CTL[2]	EAC[3]
Data Parity		A DC data parity error for a tag hit occurred during an LS access to the DC.	DDP	E

Table 180: MC0 Error Descriptions (Continued)

Error Type	Error Sub-type	Description[1]	CTL[2]	EAC[3]
Tag Parity		A DC tag array parity error for a data hit occurred during an LS access to the DC.	DTP	E
Copyback Parity		A DC data or tag array parity error for a tag miss occurred during an LS access to the DC.	DDP, DTP	E
Tag Snoop Parity		A tag parity error was encountered during snoop. Data array parity check enabled only for tag hit.	DDP, DTP	E
L2 TLB Error	TLB parity	Parity error in BTLB. Regardless of whether this error is masked from logging, it invalidate the lookup index (for both 4K and 2M storage) where the error occurred and an L2DTLB miss is generated.	TLB2P	E
L1 TLB Error	TLB multimatch	Hit multiple entries. Regardless of whether this error is masked from logging or enabled for Sync flood generation, the L1DTLB is flushed.	TLB1M	E
L2 TLB Error	TLB multimatch	Hit multiple entries. Regardless of whether this error is masked from logging, it invalidates the lookup index (for both 4K and 2M storage) where the error occurred and an L2DTLB miss is generated.	TLB2M	E
System Read Data Error	TLB Reload	System read data error occurred on a TLB reload.	SRDET	E
	Store	System read data error occurred on a store.	SRDES	E
	Load	System read data error occurred on a load.	SRDEL	E

1. CID: core ID. All LS errors are reported to the affected core; see 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
2. See MSR0000_0400.
3. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled. See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].

Table 181: MC0 Error Signatures

Error Type	Error	ErrorCodeExt	Error Code						UC	ADDRV	PCC
			Type	UU/PP	T	RRRR	II/TT	LL			
Data Parity	Load,	0h	MEM	-	-	DRD	Data	L1	1	1	0
	Store	0h		-	-	DWR	Data	L1	1	1	1
Tag Parity	Load,	10h		-	-	DRD	Data	L1	1	1	1
	Store	10h		-	-	DWR	Data	L1	1	1	1
Copyback Parity	-	0h		-	-	Evict	Data	L1	1	1	1
Tag Snoop Parity	-	10h		-	-	Snoop	Data	L1	1	1	1
L2DTLB Parity	-	0h	TLB	-	-	-	Data	L2	0	1	0
L1DTLB Multimatch	-	1h		-	-	-	Data	L1	0[1]	1	0[1]
L2DTLB Multimatch	-	1h		-	-	-	Data	L2	0	1	0
Read Data Error on TLB Reload	-	0h	BUS	SRC	0	RD	MEM/IO	LG	1	1	1
Read Data Error on Store	-	0h		SRC	0	DWR	MEM/IO	LG	1	1	1
Read Data Error on Load	-	0h		SRC	0	DRD	MEM/IO	LG	1	1	1

1. If an MCA error causes an unlocking store to miss in both the L1DTLB and L2DTLB, the error is uncorrectable. Both PCC and UC are set to '1'.

Table 182: MC4 Error Descriptions

Error Type	Description	CTL[1]	ETG[2]	EAC[4]
Sync Error	Link-defined sync error packets detected on link. The NB floods its outgoing links with sync packets after detecting a sync packet on an incoming link independent of the state of the control bits.	SyncPktEn	L	D
Master Abort	Master abort seen as result of link operation. Reasons for this error include requests to non-existent addresses. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of the control bit.	MstrAbortEn	L	D
Target Abort	Target abort seen as result of link operation. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of the control bit.	TgtAbortEn	L	D
RMW Error	An atomic read-modify-write (RMW) command was received from an IO link. Atomic RMW commands are not supported. An atomic RMW command results in a link error response being generated back to the requesting IO device. The generation of the link error response is not affected by the control bit.	AtomicRMWEn	L	D

Table 182: MC4 Error Descriptions (Continued)

Error Type	Description	CTL[1]	ETG[2]	EAC[4]
WDT Error	NB WDT timeout due to lack of progress. The NB WDT monitors transaction completions. A transaction that exceeds the programmed time limit reports errors via the MCA. The cause of error may be another node or device which failed to respond.	WDTRptEn	L	D
DRAM ECC Error	A DRAM ECC error detected.	CECCEn, UECCEn	D	D
Link Data Error	Data error detected on link. If enabled for reporting and the request is sourced from a core, then PCC is set. (If not enabled for reporting, PCC is not set. If configured to allow an error response to be returned to the core, this could allow error containment to a scope smaller than the entire system.)	McaUsPwDatErrEn, CpPktDatEn	L	D
Protocol Error	Protocol error detected by link. These errors are distinguished from each other by the value in {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]}. See Table 184 [MC4 Error Signatures, Part 2]. For protocol errors, the system cannot continue operation. Protocol errors can be caused by other subcomponents than the one reporting the error. For diagnosis, collect and examine MCA registers from other banks, cores, and processors in the system.	NbIntProtEn	L[3]	D
NB Array Error	A parity error was detected in the NB internal arrays.	NbArrayParEn	-	D
DRAM Addr/CMD Parity Error	A parity error was detected in the DRAM address or control signals.	DramParEn	D	D
Write DRAM CRC error	A write DRAM CRC error detected.	McaWrCrcErrEn	D	D
Secure Memory Violation	Access violation to Secure Memory Region. See D18F3x2E0[On3SecMemErrEn].	On3SecMemErrEn	D	D

Table 182: MC4 Error Descriptions (Continued)

Error Type	Description	CTL[1]	ETG[2]	EAC[4]
Compute Unit Data Error	NB received a data error from a core and this error could not be contained. For the cause of the data error, examine the core MCA registers for deferred errors. This error may occur for the following types of data writes: • APIC • Configuration space (IO and MMIO) For these errors, Sync flood occurs if D18F3x180[SyncFloodOnCpuLeakErr] is set.	McaCpuDatErrEn	-	D

1. CTL: See MSR0000_0410.
2. ETG: error threshold group. See 2.16.1.7 [Error Thresholding].
 - L=Link.
 - D=DRAM.
3. The error thresholding group is Link if link protocol error; none for non-link protocol error.
4. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled.

The NB is capable of reporting the following errors (See Table 60 [Error Code Types] - Table 66 [Error Codes: Internal Error Type (UU)]):

Table 183: MC4 Error Signatures, Part 1

Error Type	ErrorCode-Ext	Error Code					
		Type	PP	T	RRRR	II/TT	LL
Reserved	00h	-	-	-	-	-	-
CRC Error	01h	BUS	OBS	0	GEN	GEN	LG
Sync Error	02h	BUS	OBS	0	GEN	GEN	LG
Mst Abort	03h	BUS	SRC/OBS	0	RD/WR	MEM/IO[1]	LG
Tgt Abort	04h	BUS	SRC/OBS	0	RD/WR	MEM/IO[1]	LG
Reserved	05h	-	-	-	-	-	-
RMW Error	06h	BUS	OBS	0	GEN	IO	LG
WDT Error	07h	BUS	GEN	1	GEN	GEN	LG
Reserved	08h	-	-	-	-	-	-
Reserved	09h	-	-	-	-	-	-
Link Data Error	0Ah	BUS	SRC/OBS	0	RD/WR/ DWR	MEM/IO	LG
NB Protocol Error	0Bh		OBS	0	GEN	GEN	LG
NB Array Error	0Ch		OBS	0	GEN	GEN	LG
DRAM Addr/CMD Parity Error	0Dh	BUS	OBS	0	GEN	MEM	LG
Retry	0Eh	BUS	OBS	1	GEN	GEN	LG
Reserved	0Fh	-	-	-	-	-	-
Reserved	10h	-	-	-	-	-	-
DDR4 Write CRC error	11h	BUS	OBS	0	WR	MEM	LG
Secure Memory Violation	12h	BUS	OBS	0	RD/WR	MEM	LG
Reserved	18h-13h	-	-	-	-	-	-
Compute Unit Data Error	19h	MEM	-	-	WR	Data	LG

Table 183: MC4 Error Signatures, Part 1 (Continued)

Error Type	ErrorCode-Ext	Error Code					
		Type	PP	T	RRRR	II/TT	LL
P2P Atomic	1Ah	BUS	OBS	0	GEN	IO	LG
Reserved	1Bh	-	-	-	-	-	-
Reserved	1Fh-1Ch	-	-	-	-	-	-

1. Indicates the type of link attached to the reporting NB, not the instruction type. MEM indicates coherent link, IO indicates IO link.

Table 184: MC4 Error Signatures, Part 2

Error Type	UC	AddrV	PCC	Syndrome Valid	CECC	UECC	Deferred	Scrub	Link	Err CoreId
Sync Error	1	0	1	-	0	0	0	0	Y	-
Mst Abort	1	1	Core[5]	-	0	0	0	0	Y	Y
Tgt Abort	1	1	Core[5]	-	0	0	0	0	Y	Y
RMW Error	1	1	0	-	0	0	0	0	Y	-
WDT Error	1	0[1] 1[2]	1	-	0	0	0	0	-	-
ECC Error	0/1	1	0/1	15:0	0/1[13]	0/1[13]	0	1/0	-	-
ECC Error, Deferred	0	1	0	15:0	0/1[13]	0/1[13]	1	1/0	-	-
Link Data Error	~Deferred	1	0	-	0	0	0/1	0	Y	-
NB Protocol Error	1	1/0[2]	1	-	0	0	0	0	Y	-
NB Array Error	~Deferred	1	~Deferred	-	0	0	0/1	0	-	-
Compute Unit Data Error	1	0	1	-	0	0	0	0	-	Y

1. See Table 190 [Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Watchdog Timer Errors].
2. See Table 186 [Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Protocol Errors].
4. See Table 189 [Valid Values for ArrayErrorType].
10. Core: source is core. 1=Source is core. 0=Source is not core.
13. Hist: Error was detected by the hardware-managed history scheme.

D18F3x50 MCA NB Address Low

IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSE Read-write; Per-node; Not-same-for-all. ENDIF. Cold reset: 0000_0000h. See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture.

MSR0000_0412[31:0] is an alias of D18F3x50.
Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products].
ErrAddr[47:1] = {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} carries supplemental information associated with a machine check error, generally the address being accessed. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. The format of ErrAddr[47:1] is a function of D18F3x48[ErrorCodeExt]; See ErrAddr[47:1].

Bits	Description
31:0	MSR0000_0412[31:0] is an alias of D18F3x50. See MSR0000_0412[31:0].

D18F3x54 MCA NB Address High

IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSE Read-write; Per-node; Not-same-for-all. ENDIF. Cold reset: 0000_0000h. See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture.
MSR0000_0412[63:32] is an alias of D18F3x54.
Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products].
ErrAddr[47:1] = {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} carries supplemental information associated with a machine check error, generally the address being accessed. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. The format of ErrAddr[47:1] is a function of D18F3x48[ErrorCodeExt]; See ErrAddr[47:1].

Bits	Description
31:0	MSR0000_0412[63:32] is an alias of D18F3x54. See MSR0000_0412[63:32].

The register format depends on the type of error being logged:
- Protocol errors contain the error reason code, may contain the physical address, and are formatted according to Table 186.
- NB array errors indicate the array in error, and are formatted according to Table 188.
- NB Watchdog timer errors depend on the mode selected by D18F3x180[McaLogErrAddrWdtErr], and the format is indicated by D18F3x4C[AddrV]. If D18F3x4C[AddrV] is indicated, errors are formatted according to Table 185. If D18F3x4C[AddrV] is not indicated, errors are formatted according to Table 190.
- All other NB errors which indicate D18F3x4C[AddrV] are formatted according to Table 185.

Table 185: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for All Other Errors

Bits	Description
47:1	**PhysAddr[47:1]**.

Table 186: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Protocol Errors

Bits	Description
47:6	**PhysAddr[47:6]**. Valid if (D18F3x4C[AddrV] == 1); otherwise reserved.
5:1	**ProtocolErrorType**. See Table 187 [Valid Values for ProtocolErrorType].

Table 187: Valid Values for ProtocolErrorType

Bits	Description
00h	Link: SRQ Read Response without matching request.
01h	Link: Probe Response without matching request.

Table 187: Valid Values for ProtocolErrorType (Continued)

Bits	Description
02h	Link: TgtDone without matching request.
03h	Link: TgtStart without matching request.
04h	Link: On3 upstream command packet buffer overflow.
05h	Link: On3 upstream packet buffer overflow.
06h	Link: Link retry packet count acknowledge overflow.
07h	Link: On3 upstream data command in the middle of a data transfer, or a data command doesn't align with a data packet.
08h	Link: Upstream extension packet followed by a packet other than a command with address; A command without the address extension packet; The command bus bits[5:0] is not the extension/NOP/Sync command encodings; Some other basic command misalignments.
09h	Link: A specific coherent-only packet from a CPU was issued to an IO link.
0Ah	Link: An invalid On3 upstream command encoding (e.g., RdBlkM, ...).
0Bh	Link: An upstream data valid occurred when a data phase was not pending.
1Fh-0Ch	Reserved.

Table 188: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for NB Array Errors

Bits	Description
47:6	Reserved.
5:1	**ArrayErrorType**. See Table 189 [Valid Values for ArrayErrorType].

Table 189: Valid Values for ArrayErrorType

Bits	Description
00h	SRA: System request address.
01h	SRD: System request data.
02h	SPB: System packet buffer.
03h	MCD: Memory controller data.
04h	MPB: Memory packet buffer.
05h	LPB0: Link 0 packet buffer.
08h-06h	Reserved.
09h	MPBC: Memory controller command packet buffer.
0Ah	MCDBM: Memory controller byte mask.
0Bh	MCACAM: Memory controller address array.
0Ch	DMAP: Extended DRAM address map.
0Dh	MMAP: Extended MMIO address map.
0Eh	X86MAP: Extended PCI/IO address map.
0Fh	CFGMAP: Extended config address map.
17h-10h	Reserved.

Table 189: Valid Values for ArrayErrorType (Continued)

Bits	Description
18h	SRIMCTRTE: SRI/MCT extended routing table.
1Ch-19h	Reserved.
1Dh	TCB: TCB array.
1Fh-1Eh	Reserved.

Table 190: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Watchdog Timer Errors

Bits	Description
47	Reserved.
46	**DstUnit[2]: destination unit[2]**. See: DstUnit[1:0].
45	**SrcUnit[2]: source unit[2]**. See: SrcUnit[1:0].
44	Reserved.
43:40	**CoreId**. Indicates the core ID if the SourcePointer specifies Core. Bits: 07h-00h — Description: CoreId
39:36	**SystemResponseCount**. This field records unspecified, implementation-specific information.
35:31	**WaitCode**. records unspecified, implementation-specific information (all zeroes means no waiting condition).
30	**WaitForPostedWrite**.
29:27	**DestinationNode**. Records the Node ID of the node addressed by the transaction.
26:25	**DstUnit[1:0]: destination unit[1:0]**. DstUnit[2:0] = {DstUnit[2], DstUnit[1:0]}. Bits — Description 000b — Core 001b — Extended Core 010b — Memory Controller 011b — Host 100b — ONION3 Link0 101b — ONION3 Link1 111b-110b — Reserved
24:22	**SourceNode**. Records the Node ID of the node originating the transaction.
21:20	**SrcUnit[1:0]: source unit[1:0]**. SrcUnit[2:0] = {SrcUnit[2], SrcUnit[1:0]}. See: DstUnit[1:0].
19:15	**SourcePointer**. Identifies crossbar source: Bits — Description 00h — SRI HostBridge. 03h-01h — Reserved. 04h — Core. See CoreId. 07h-05h — Reserved. 08h — Memory controller. 0Fh-09h — Reserved. 1Fh-10h — Link. Link HH; sublink N (where N == 0b for ganged links). All unused codes are reserved.

Table 190: Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Watchdog Timer Errors (Continued)

Bits	Description
14:11	**SrqEntryState**. Records unspecified, implementation-specific information (all zeroes means idle).
10:7	**OpType**. Records unspecified, implementation-specific information.
6:1	**LinkCommand**. When the NB WDT expires, the link command of the transaction that timed out is captured here. This field is encoded identically to the "Code" field for link transactions defined in the link specification.

D18F3x64 Hardware Thermal Control (HTC)

See 2.10.3.1 [PROCHOT_L and Hardware Thermal Control (HTC)]. If D18F3xE8[HtcCapable] == 0 then this register is reserved.

Bits	Description
31:0	Alias of D0F0xBC_xD820_0C64.

D18F3x68 Software P-state Limit

See 2.10.3.2 [Software P-state Limit Control]. If D18F3xE8[HtcCapable] == 0 then this register is reserved.

Bits	Description
31	Reserved.
30:28	**SwPstateLimit: software P-state limit select**. Read-write. Reset: Product-specific. Specifies a P-state limit for all cores. Uses hardware P-state numbering; see 2.5.2.1.1.2 [Hardware P-state Numbering]. Not changed on a write if the value written is greater than D18F3xDC[HwPstateMaxVal] or less than D18F4x15C[NumBoostStates]. See SwPstateLimitEn.
27:6	Reserved.
5	**SwPstateLimitEn: software P-state limit enable**. Read-write. Reset: 0. 1=SwPstateLimit is enabled.
4:0	Reserved.

D18F3x6C Data Buffer Count

Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

- To ensure deadlock free operation the following minimum buffer allocations are required:
 - D18F3x6C[UpRspDBC] >= 1.
 - D18F3x6C[DnReqDBC] >= 1.
 - D18F3x6C[UpReqDBC] >= 1.
 - D18F3x6C[DnRspDBC] >= 1.
- If D18F0x[E4,C4,A4,84][IsocEn] == 1: IsocRspDBC >= 1.
- The total number of data buffers allocated in this register and D18F3x7C must satisfy the following equation:

- D18F3x6C[UpReqDBC] + D18F3x6C[UpRspDBC] + D18F3x6C[DnReqDBC] + D18F3x6C[DnRspDBC] + D18F3x6C[IsocRspDBC] + (IF (D18F3x7C[Sri2XbarFreeRspDBC] == 0) THEN (D18F3x7C[Sri2XbarFreeXreqDBC]*2). ELSE D18F3x7C[Sri2XbarFreeXreqDBC]. ENDIF.) + D18F3x7C[Sri2XbarFreeRspDBC] <= 16.

Bits	Description
31	Reserved.
30:28	**IsocRspDBC: isochronous response data buffer count**. Read-write. Reset: 1. BIOS: 2.
27:19	Reserved.
18:16	**UpRspDBC: upstream response data buffer count**. Read-write. Reset: 1. BIOS: 2.
15:8	Reserved.
7:6	**DnRspDBC: downstream response data buffer count**. Read-write. Reset: 1.
5:4	**DnReqDBC: downstream request data buffer count**. Read-write. Reset: 1.
3	Reserved.
2:0	**UpReqDBC: upstream request data buffer count**. Read-write. Reset: 1. BIOS: 2.

D18F3x70 SRI to XBAR Command Buffer Count

Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

- To ensure deadlock free operation the following minimum buffer allocations are required:
 - D18F3x70[UpRspCBC] >= 1.
 - D18F3x70[UpPreqCBC] >= 1.
 - D18F3x70[DnPreqCBC] >= 1.
 - D18F3x70[UpReqCBC] >= 1.
 - D18F3x70[DnReqCBC] >= 1.
 - D18F3x70[DnRspCBC] >= 1.
- If any of the D18F0x[E4,C4,A4,84][IsocEn] bits are set:
 IsocReqCBC >= 1 IsocRspCBC >= 1
- If D18F0x[E4,C4,A4,84][IsocEn] == 1 and isochronous posted requests may be generated by the system:
 IsocPreqCBC >= 1
- The total number of SRI to XBAR command buffers allocated in this register and D18F3x7C must satisfy the following equation:
 - D18F3x70[IsocRspCBC] + D18F3x70[IsocPreqCBC] + D18F3x70[IsocReqCBC] + D18F3x70[UpRspCBC] + D18F3x70[DnPreqCBC] + D18F3x70[UpPreqCBC] + D18F3x70[DnReqCBC] + D18F3x70[DnRspCBC] + D18F3x70[UpReqCBC] + D18F3x7C[Sri2XbarFreeRspCBC] + D18F3x7C[Sri2XbarFreeXreqCBC] <= 48.

Bits	Description
31	Reserved.
30:28	**IsocRspCBC: isoc response command buffer count**. Read-write. Reset: 1.
27	Reserved.
26:24	**IsocPreqCBC: isoc posted request command buffer count**. Read-write. Reset: 1. BIOS: 0.
23	Reserved.
22:20	**IsocReqCBC: isoc request command buffer count**. Read-write. Reset: 1.

19	Reserved.
18:16	**UpRspCBC: upstream response command buffer count**. Read-write. Reset: 1. BIOS: 3.
15	Reserved.
14:12	**DnPreqCBC: downstream posted request command buffer count**. Read-write. Reset: 2. BIOS: 1.
11	Reserved.
10:8	**UpPreqCBC: upstream posted request command buffer count**. Read-write. Reset: 1.
7:6	**DnRspCBC: downstream response command buffer count**. Read-write. Reset: 1.
5:4	**DnReqCBC: downstream request command buffer count**. Read-write. Reset: 1.
3	Reserved.
2:0	**UpReqCBC: upstream request command buffer count**. Read-write. Reset: 3.

D18F3x74 XBAR to SRI Command Buffer Count

Reset: 0007_1111h. Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

Table 191: Buffer Definitions

Term	Definition
SpqSize	Probe command queue size. SpqSize = 32.
SrqSize	SRQ (XBAR command and probe response to SRI) queue size. SrqSize = 68.
PrbRsp	SRQ entries hard allocated to probe responses. PrbRsp = 4.
MpbcSize	MPB command buffer size. MpbcSize = 32.
McqSize	MCT command queue size. McqSize = 72.

- To ensure deadlock free operation the following minimum buffer allocations are required:
 - D18F3x74[ProbeCBC] >= 2.
 - D18F3x74[UpReqCBC] >= 1.
 - D18F3x74[UpPreqCBC] >= 1.
 - (IsocReqCBC + IsocPreqCBC + DRReqCBC) <= 31.
 - (IsocReqCBC + IsocPreqCBC + DRReqCBC) <= (McqSize - 16).
- If any of D18F0x[E4,C4,A4,84][IsocEn] bits are set, then IsocReqCBC >= 1.
- If any of the D18F0x[E4,C4,A4,84][IsocEn] bits are set and isochronous posted requests may be generated by the system:
 IsocPreqCBC >= 1
- The total number of XBAR to SRI command buffers allocated in this register and D18F3x7C must satisfy the following equation:
 - D18F3x74[UpReqCBC] + D18F3x74[UpPreqCBC] + D18F3x74[DnReqCBC] + D18F3x74[DnPreqCBC] + D18F3x74[IsocReqCBC] + D18F3x74[IsocPreqCBC] +

D18F3x74[DRReqCBC] + D18F3x7C[Xbar2SriFreeListCBC] + (D18F3x1A0[CpuCmdBufCnt] * NumOfCompUnits) + D18F3x1A0[CpuToNbFreeBufCnt] + PrbRsp <= SrqSize

- The total number of SPQ (probe command) buffers allocated must satisfy the following equation:
 - (D18F3x17C[SPQPrbFreeCBC] + D18F3x74[ProbeCBC]) <= SpqSize.

Bits	Description
31:28	**DRReqCBC: display refresh request command buffer count**. Read-write.
27	Reserved.
26:24	**IsocPreqCBC: isochronous posted request command buffer count**. Read-write. BIOS: 1.
23:20	**IsocReqCBC: isochronous request command buffer count**. Read-write. BIOS: 1.
19:16	**ProbeCBC: probe command buffer count**. Read-write. BIOS: Ch. Bits: 0h = 0 buffers; Ch-1h = <ProbeCBC> buffers; Fh-Dh = Reserved.
15	Reserved.
14:12	**DnPreqCBC: downstream posted request command buffer count**. Read-write. BIOS: 0.
11	Reserved.
10:8	**UpPreqCBC: upstream posted request command buffer count**. Read-write.
7	Reserved.
6:4	**DnReqCBC: downstream request command buffer count**. Read-write. BIOS: 0.
3	Reserved.
2:0	**UpReqCBC: upstream request command buffer count**. Read-write.

D18F3x78 MCT to XBAR Buffer Count

Reset: 0018_0513h. Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

- To ensure deadlock free operation the following minimum buffer allocations are required:
 ProbeCBC >= 1 RspCBC >= 1 RspDBC >= 2
 RspDBC >= D18F2x11C[MctPrefReqLimit]+2
- The total number of command buffers allocated in this register must satisfy the following equation:
 (D18F3x78[ProbeCBC] + D18F3x78[RspCBC]) <= MpbcSize.

Bits	Description
31:22	Reserved.
21:16	**RspDBC: response data buffer count**. Read-write. BIOS: 20h. Bits: Description 01h-00h: Reserved 02h: 2 Buffers 1Fh-03h: <RspDBC> Buffers 20h: 32 Buffers 3Fh-21h: Reserved
15:13	Reserved.
12:8	**ProbeCBC: probe command buffer count**. Read-write. BIOS: Ch.
7:6	Reserved.
5:0	**RspCBC: response command buffer count**. Read-write. BIOS: 14h.

D18F3x7C Free List Buffer Count

Reset: 0001_660Ch. Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values. See D18F3x6C and D18F3x70.

- To ensure deadlock free operation the following minimum buffer allocations are required:
 - IF (D18F3x7C[Sri2XbarFreeRspCBC] == 0) THEN (D18F3x7C[Sri2XbarFreeXreqCBC] > 2).
 - IF (D18F3x7C[Sri2XbarFreeRspCBC] != 0) THEN (D18F3x7C[Sri2XbarFreeRspCBC] > 2).
 - IF (D18F3x7C[Sri2XbarFreeRspDBC] == 0) THEN (D18F3x7C[Sri2XbarFreeXreqDBC] > 2).
 - IF (D18F3x7C[Sri2XbarFreeRspDBC] != 0) THEN (D18F3x7C[Sri2XbarFreeRspDBC] > 2).
 - D18F3x7C[Xbar2SriFreeListCBC] >= (D18F3x1A0[CpuToNbFreeBufCnt] * NumOfCompUnits) + 2.

Bits	Description
31	Reserved.
30:28	**Xbar2SriFreeListCBInc: XBAR to SRI free list command buffer increment**. Read-write. This field is use to add buffers to the free list pool if they are reclaimed from hard allocated entries without having to go through warm reset. This field may only be programmed after buffers have been allocated and released via D18F0x6C[RlsLnkFullTokCntImm].
27:23	Reserved.
22:20	**Sri2XbarFreeRspDBC: SRI to XBAR free response data buffer count**. Read-write.
19:16	**Sri2XbarFreeXreqDBC: SRI to XBAR free request and posted request data buffer count**. Read-write. BIOS: 4h. If Sri2XbarFreeRspDBC == 0h, then these buffers are shared between requests, responses and posted requests and the number of buffers allocated is two times the value of this field.

15:12	**Sri2XbarFreeRspCBC: SRI to XBAR free response command buffer count**. Read-write. BIOS: 0h.
11:8	**Sri2XbarFreeXreqCBC: SRI to XBAR free request and posted request command buffer count**. Read-write. BIOS: Ah. If Sri2XbarFreeRspCBC == 0h, then these buffers are shared between requests, responses and posted requests and the number of buffers allocated is two times the value of this field.
7:6	Reserved.
5:0	**Xbar2SriFreeListCBC: XBAR to SRI free list command buffer count**. Read-write. BIOS: 3Ah.

D18F3x[84:80] ACPI Power State Control

This block consists of eight identical 8-bit registers, one for each System Management Action Field (SMAF) code associated with STPCLK assertion commands from the link. Refer to the descriptions below for the associated ACPI state and system management actions for each of the 8 SMAF codes. The SmafAct fields specify the system management actions taken when the corresponding SMAF code is received. For instance, a SMAF code of 5 results in the power management actions specified by SmafAct5. Some ACPI states and associated SMAF codes may not be supported in certain conditions. See 2.5 [Power Management] for which states are supported.

When a link STPCLK assertion command is received by the processor, the power management commands specified by the register with the corresponding SMAF code are invoked. When the STPCLK deassertion command is received by the processor, the processor returns into the operational state.

In multi-node systems, these registers should be programmed identically in all nodes.

Table 192: SMAF Action Definition

Register	SmafAct	ACPI state	Description
D18F3x84[31:24]	SmafAct7	C1	Initiated when a Halt instruction is executed by processor. This does not involve the interaction with the SMC, therefore the SMC is required to never send STPCLK assertion commands with SMAF == 7h.
D18F3x84[23:16]	SmafAct6	S4/S5	Initiated by a processor access to the ACPI-defined PM1_CNTa register.
D18F3x84[15:8]	SmafAct5	Throttling	Occurs based upon SMC hardware-initiated throttling. AMD recommends using PROCHOT_L for thermal throttling and not implementing stop clock based throttling.
D18F3x84[7:0]	SmafAct4	S3	Initiated by a processor access to the ACPI-defined PM1_CNTa register.
D18F3x80[31:24]	SmafAct3	S1	Initiated by a processor access to the ACPI-defined PM1_CNTa register.
D18F3x80[23:16]	SmafAct2	-	
D18F3x80[15:8]	SmafAct1	C1E, or Link init.	Initiated by an access to the ACPI-defined P_LVL3 register.
D18F3x80[7:0]	SmafAct0	C2	Initiated by a processor access to the ACPI-defined P_LVL2 register.

D18F3x80 ACPI Power State Control Low

Reset: 0000_0000h. Read-write.

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31:29</td><td>ClkDivisorSmafAct3. See: ClkDivisorSmafAct0.</td></tr>
<tr><td>28:27</td><td>Reserved.</td></tr>
<tr><td>26</td><td>NbGateEnSmafAct3. See: NbGateEnSmafAct0.</td></tr>
<tr><td>25</td><td>NbLowPwrEnSmafAct3. See: NbLowPwrEnSmafAct0.</td></tr>
<tr><td>24</td><td>CpuPrbEnSmafAct3. See: CpuPrbEnSmafAct0.</td></tr>
<tr><td>23:21</td><td>ClkDivisorSmafAct2. See: ClkDivisorSmafAct0.</td></tr>
<tr><td>20:19</td><td>Reserved.</td></tr>
<tr><td>18</td><td>NbGateEnSmafAct2. See: NbGateEnSmafAct0.</td></tr>
<tr><td>17</td><td>NbLowPwrEnSmafAct2. See: NbLowPwrEnSmafAct0.</td></tr>
<tr><td>16</td><td>CpuPrbEnSmafAct2. See: CpuPrbEnSmafAct0.</td></tr>
<tr><td>15:13</td><td>ClkDivisorSmafAct1. See: ClkDivisorSmafAct0.</td></tr>
<tr><td>12:11</td><td>Reserved.</td></tr>
<tr><td>10</td><td>NbGateEnSmafAct1. See: NbGateEnSmafAct0.</td></tr>
<tr><td>9</td><td>NbLowPwrEnSmafAct1. See: NbLowPwrEnSmafAct0.</td></tr>
<tr><td>8</td><td>CpuPrbEnSmafAct1. See: CpuPrbEnSmafAct0.</td></tr>
<tr><td>7:5</td><td>ClkDivisorSmafAct0: clock divisor. Read-write.
Specifies the core clock frequency while in the low-power state. This divisor is relative to the current FID frequency, or:
• 100 MHz * (10h + MSRC001_00[6B:64][CpuFid[5:0]]) of the current P-state specified by MSRC001_0063[CurPstate].
If MSRC001_00[6B:64][CpuDid] of the current P-state indicates a divisor that is deeper than specified by this field, then no frequency change is made when entering the low-power state associated with this register.
<table>
<tr><th>Bits</th><th>Description</th><th>Bits</th><th>Description</th></tr>
<tr><td>000b</td><td>/1</td><td>100b</td><td>/16</td></tr>
<tr><td>001b</td><td>/2</td><td>101b</td><td>/128</td></tr>
<tr><td>010b</td><td>/4</td><td>110b</td><td>/512</td></tr>
<tr><td>011b</td><td>/8</td><td>111b</td><td>Turn off clocks</td></tr>
</table>
</td></tr>
<tr><td>4:3</td><td>Reserved.</td></tr>
</table>

2	**NbGateEnSmafAct0: Northbridge gate enable**. Read-write. This bit does not control hardware. NbLowPwrEn is required to be set if this bit is set.
1	**NbLowPwrEnSmafAct0: Northbridge low-power enable**. Read-write. 1=The NB clock is ramped down to the divisor specified by D18F3xD4[NbClkDiv] and DRAM is placed into self-refresh mode when LDTSTOP_L is asserted while in the low-power state.
0	**CpuPrbEnSmafAct0: CPU direct probe enable**. Read-write. Specifies how probes are handled while in the low-power state. 0=When the probe request comes into the NB, the core clock is brought up to the COF (based on the current P-state), all outstanding probes are completed, the core waits for a hysteresis time based on D18F3xD4[ClkRampHystSel], and then the core clock is brought down to the frequency specified by ClkDivisor. 1=The core clock does not change frequency; the probe is handled at the frequency specified by ClkDivisor; this may only be set if: • ClkDivisor specifies a divide by 1, 2, 4, 8, or 16 and NbCof <= 3.2 GHz • ClkDivisor specifies a divide by 1, 2, 4, or 8 and NbCof >= 3.4 GHz This bit also specifies functionality of the timer used for cache flushing during halt. See D18F3xDC[CacheFlushOnHaltTmr]. • If ((D18F3x84[CpuPrbEnSmafAct7] == 0) && (D18F3xDC[IgnCpuPrbEn] == 0)), only the time when the core is halted and has its clocks ramped up to service probes is counted. • If ((D18F3x84[CpuPrbEnSmafAct7] == 1) or (D18F3xDC[IgnCpuPrbEn] == 1)), all of the time the core is halted is counted.

D18F3x84 ACPI Power State Control High

Reset: 0000_0000h. Read-write.

Bits	Description
31:29	**ClkDivisorSmafAct7**. See: D18F3x80[ClkDivisorSmafAct0].
28:27	Reserved.
26	**NbGateEnSmafAct7**. See: D18F3x80[NbGateEnSmafAct0].
25	**NbLowPwrEnSmafAct7**. See: D18F3x80[NbLowPwrEnSmafAct0].
24	**CpuPrbEnSmafAct7**. See: D18F3x80[CpuPrbEnSmafAct0].
23:21	**ClkDivisorSmafAct6**. See: D18F3x80[ClkDivisorSmafAct0].
20:19	Reserved.
18	**NbGateEnSmafAct6**. See: D18F3x80[NbGateEnSmafAct0].
17	**NbLowPwrEnSmafAct6**. See: D18F3x80[NbLowPwrEnSmafAct0].
16	**CpuPrbEnSmafAct6**. See: D18F3x80[CpuPrbEnSmafAct0].
15:13	**ClkDivisorSmafAct5**. See: D18F3x80[ClkDivisorSmafAct0].
12:11	Reserved.
10	**NbGateEnSmafAct5**. See: D18F3x80[NbGateEnSmafAct0].
9	**NbLowPwrEnSmafAct5**. See: D18F3x80[NbLowPwrEnSmafAct0].
8	**CpuPrbEnSmafAct5**. See: D18F3x80[CpuPrbEnSmafAct0].
7:5	**ClkDivisorSmafAct4**. See: D18F3x80[ClkDivisorSmafAct0]. BIOS: 111b.
4:3	Reserved.

2	**NbGateEnSmafAct4**. See: D18F3x80[NbGateEnSmafAct0].
1	**NbLowPwrEnSmafAct4**. See: D18F3x80[NbLowPwrEnSmafAct0]. BIOS: 1.
0	**CpuPrbEnSmafAct4**. See: D18F3x80[CpuPrbEnSmafAct0].

D18F3x88 NB Configuration 1 Low (NB_CFG1_LO)

Bits	Description
31	**DisCohLdtCfg: disable coherent link configuration accesses**. Read-write; Per-node. Reset: 0. 1=Disables automatic routing of PCI configuration accesses to the processor configuration registers; PCI configuration space accesses which fall within the hard-coded range reserved for processor configuration-space registers are instead routed to the IO link specified by D18F1x[1DC:1D0,EC:E0] [Configuration Map]. This can be used to effectively hide the configuration registers from software. It can also be used to provide a means for an external chip to route processor configuration accesses according to a scheme other than the hard-coded version. When used, this bit needs to be set on all processors in a system. PCI configuration accesses should not be generated if this bit is not set on all processors.
30:19	Reserved.
18	**DisCstateBoostBlockPstateUp**. Read-write. Reset: 0. BIOS: 1. 1=Allow cores that are waking up out of a non-C0 C-state to transition to the last requested Pstate without having to wait for cores in the boosted P-state to transition out of the boosted P-state.
17:0	Reserved.

D18F3x8C NB Configuration 1 High (NB_CFG1_HI)

Bits	Description
31	Reserved.
30	**DisStpClkAbortFlush**. Read-write; Per-node. Reset: X. 1=Disable aborting flush for core when the other core has a pending architectural interrupt.
29:24	Reserved.
23	**EnaDiv1CpuLowPwr**. Read-write; Per-node. Reset: X. Enables power management actions in the core even when the requested clock divisor is /1. Normally a /1 clock divisor does not generate power management actions.
22	**InitApicIdCpuIdLo**. Read-write; Per-node. Reset: X. BIOS: 1. 0=Reserved. 1=Selects the format for ApicId; see APIC20.
21:20	Reserved.
19	**DisDatFwdVic**. Read-write; Per-node. Reset: 1. 1=Disables data forwarding from victims to reads.
18	**DisOrderRdRsp**. Read-write; Per-node. Reset: X. 1=Disables ordered responses to IO link read requests.
17:15	Reserved.
14	**EnableCf8ExtCfg: enable CF8 extended configuration cycles**. Read-write; Per-node. Reset: 0. 1=Allows the IO configuration space access method, IOCF8 and IOCFC, to be used to generate extended configuration cycles by enabling IOCF8[27:24].

13	**DisUsSysMgtReqToNcHt: disable upstream system management request to link**. Read-write; Per-node. Reset: X. 1=Disables downstream reflection of upstream STPCLK and x86 legacy input system management commands (in order to work around potential deadlock scenarios related to reflection regions).
12:5	Reserved.
4	**DisDatMsk: disable data mask**. Read-write; Per-node. Reset: 0. BIOS: IF (DataMaskMbType != 1) THEN 1. ELSE 0. ENDIF. 1=Disables DRAM data masking function; all write requests that are less than one cacheline, a DRAM read is performed before writing the data.
3:0	Reserved.

D18F3xA0 Power Control Miscellaneous

Bits	Description
31:0	Reserved.

D18F3xA8 Pop Up and Down P-states

Bits	Description
31:29	**PopDownPstate**. Read-write. Reset: D18F3xDC[HwPstateMaxVal]. Specifies the pop-down P-state number. This field uses hardware P-state numbering. See 2.5.2.2.3.3 [Core C6 (CC6) State].
28:0	Reserved.

D18F3xB8 NB Array Address

Reset: XXXX_XXXXh. D18F3xB8 [NB Array Address] and D18F3xBC [NB Array Data Port] provide a mechanism to inject errors into DRAM and data read from internal NB arrays.

D18F3xB8 should first be written with the target array and address within the array. Read and write accesses to D18F3xBC then access the target address within the target array.

Bits	Description
31:28	**ArraySelect**. Read-write. Selects the NB array to access. Bits Description Fh-0h Reserved.
27:10	Reserved.
9:0	**ArrayAddress**. Read-write. Selects the location to access within the selected array. This format of this field is a function of ArraySelect. ArraySelect Description 3FFh-000h Reserved.

D18F3xBC NB Array Data Port

See D18F3xB8 for register access information. Address: D18F3xB8[ArraySelect].

Bits	Description
31:0	**Data**.

D18F3xC0 COFVID Control

Cold reset: Product-specific.

Bits	Description
31:24	**NbVid: Northbridge VID**. IF (MSRC001_0071[NbPstateDis]) THEN Read-only. ELSE Read-write. ENDIF. See D18F5x16[C:0][NbVid].
23	RAZ.
22	**NbPstate: Northbridge P-state**. IF (MSRC001_0071[NbPstateDis]) THEN Read-only. ELSE Read-write. ENDIF. See MSRC001_00[6B:64][NbPstate].
21	RAZ.
20	**CpuVid[7]**. Read-write. See CpuVid[6:0].
19	RAZ.
18:16	**PstateId: P-state identifier**. Read-write. This field is required to provide the P-state number that is associated with the values of the other fields in this register. This value is used by the logic to determine if the P-state is increasing or decreasing. This field uses hardware P-state numbering. See 2.5.2.1.1.2 [Hardware P-state Numbering].
15:9	**CpuVid[6:0]: core VID**. Read-write. See MSRC001_00[6B:64][CpuVid]. CpuVid[7:0] = {CpuVid[7], CpuVid[6:0]}.
8:6	**CpuDid: core divisor ID**. Read-write. See MSRC001_00[6B:64][CpuDid]. The PstateId field must be updated to cause a new CpuDid value to take effect.
5:0	**CpuFid[5:0]: core frequency ID**. Read-write. See MSRC001_00[6B:64][CpuFid]. The PstateId field must be updated to cause a new CpuFid value to take effect.

D18F3xC4 SBI P-state Limit

Bits	Description
31:11	Reserved.
10:8	**PstateLimit: P-state limit select**. Read-only. Reset: 0. Uses hardware P-state numbering.
7:1	Reserved.
0	**PstateLimitEn: P-state limit enable**. Read-only. Reset: 0. 1=PstateLimit is enabled.

D18F3xC8 COFVID Status Low

See 2.5.2 [CPU Core Power Management].

Bits	Description
31:24	**CurNbVid[7:0]: current NB VID**. Read-only; Updated-by-hardware. Cold reset: Product-specific. This field specifies the current VDDNB voltage.
23	**NbPstateDis: NB P-states disabled**. Value: D18F5x174[NbPstateDis]. MSRC001_0071[NbPstate-Dis] is an alias of D18F5x174[NbPstateDis]. 0=NB P-state frequency and voltage changes are supported. See D18F5x170[SwNbPstateLoDis, NbPstateDisOnP0]. 1=NB P-state frequency and voltage changes are disabled.
22:21	Reserved.
20	**CurCpuVid[7]**. Read-only; Updated-by-hardware; Not-same-for-all. Cold reset: Product-specific. See CurCpuVid[6:0].
19	Reserved.
18:16	**CurPstate: current P-state**. Read-only; Updated-by-hardware; Not-same-for-all. Cold reset: Prod-uct-specific. Specifies the current P-state requested by the core. This field uses hardware P-state num-bering. See MSRC001_0063[CurPstate] and 2.5.2.1.1.2 [Hardware P-state Numbering]. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed.
15:9	**CurCpuVid[6:0]: current core VID**. Read-only; Updated-by-hardware; Not-same-for-all. Cold reset: Product-specific. CurCpuVid = {CurCpuVid[7], CurCpuVid[6:0]}. This field specifies the current VDD voltage.
8:6	**CurCpuDid: current core divisor ID**. Read-only; Updated-by-hardware. Cold reset: Product-spe-cific. Specifies the current CpuDid of the core. See MSRC001_00[6B:64]. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed.
5:0	**CurCpuFid: current core frequency ID**. Read-only; Updated-by-hardware. Cold reset: Product-specific. Specifies the current CpuFid of the core. See MSRC001_00[6B:64]. When a P-state change is requested, the value in this field is updated once all required frequency and voltage transitions are completed.

D18F3xCC COFVID Status High

See 2.5.2 [CPU Core Power Management].

Bits	Description
31:27	**MaxNbCof[4:0]: maximum NB COF**. Read-only. Cold reset: Product-specific. MaxNbCof = {MaxNbCof[5], MaxNbCof[4:0]}. Specifies the maximum NB COF supported by the processor. If MaxNbCof is greater than zero, the maximum frequency is 100 MHz * MaxNbCof; if MaxNbCof = 00h, then there is no frequency limit. Any attempt to change the NB COF to a frequency greater than specified by this field is ignored.
26:24	**CurPstateLimit: current P-state limit**. Read-only; Updated-by-hardware. Provides the current highest-performance P-state limit number. This register uses hardware P-state numbering. See MSRC001_0061[CurPstateLimit]and 2.5.2.1.1.2 [Hardware P-state Numbering].

23	Reserved.
22:17	**MaxCpuCof: maximum core COF**. Read-only. Cold reset: Product-specific. Specifies the maximum CPU COF supported by the processor. The maximum frequency is 100 MHz * MaxCpuCof, if MaxCpuCof is greater than zero; if MaxCpuCof == 00h, then there is no frequency limit. Any attempt to change a CPU COF to a frequency greater than specified by this field is ignored.
16:4	Reserved.
3	**MaxNbCof[5]: maximum NB COF**. Read-only. Cold reset: Product-specific. See MaxNbCof[4:0].
2:0	**StartupPstate: startup P-state number**. Read-only. Cold reset: Product-specific. Specifies the cold reset VID, FID and DID for the core based on the P-state number selected. StartupPstate uses hardware P-state numbering. See MSRC001_00[6B:64] and 2.5.2.1.1.2 [Hardware P-state Numbering].

D18F3xD4 Clock Power/Timing Control 0

Bits	Description
31	**NbClkDivApplyAll**. Read-write. Cold reset: 0. BIOS: 1. See NbClkDiv.
30:28	**NbClkDiv: NB clock divisor**. Read-write. Cold reset: Product-specific. BIOS: 100b. Specifies the NB CLK divisor associated with D18F3x80/D18F3x84[NbLowPwrEn]. This divisor is applied while LDTSTOP_L is asserted if the corresponding core CLK divisor, D18F3x80/D18F3x84[ClkDivisor], is set to "turn off clocks" or if NBClkDivApplyAll == 1; otherwise, the divisor specified by D18F3x80/D18F3x84[ClkDivisor] is applied. This divisor is relative to the current NB FID frequency, or: 100 MHz * (4 + D18F5x16[C:0][NbFid[5:0]]). If D18F5x16[C:0][NbDid] of the current NB P-state indicates a divisor that is lower than specified by this field, then no NB frequency change is made when entering the low-power state associated with this register (i.e., if this field specifies a divide by 1 and the DID is divide by 2, then the divisor remains 2 while in the low-power state). This field is encoded as follows: Bits / Description / Bits / Description 000b / Divide by 1 / 100b / Divide by 16 001b / Divide by 2 / 101b / Reserved. 010b / Divide by 4 / 110b / Reserved. 011b / Divide by 8 / 111b / Reserved.
27:18	Reserved.
17:15	Reserved.
14	**CacheFlushImmOnAllHalt: cache flush immediate on all halt**. Read-write. Cold reset: 0. 1=Flush the caches immediately when all cores in a package have halted. The following condition must be true in order for the caches to be flushed: • D18F4x118/D18F4x11C[CacheFlushEn] = 1 for the corresponding C-state action field on all cores.
13	Reserved.
12	**ClkRampHystCtl: clock ramp hysteresis control**. Read-write. Cold reset: 0. Specifies the time base for ClkRampHystSel when (D18F4x128[CoreCstateMode] ? (D18F3x80/D18F3x84[CpuPrbEnSmafAct] = 0) : (D18F4x118/D18F4x11C[CpuPrbEnCstAct] = 0)). 0=320 ns. 1=1.28 us.

11:8	**ClkRampHystSel: clock ramp hysteresis select**. Read-write. Cold reset: 0h. BIOS: Fh. When the core(s) are in the stop-grant or halt state and a probe request is received, the core clock may need to be brought up to service the probe. • If (D18F4x128[CoreCstateMode] ? (D18F3x80/D18F3x84[CpuPrbEnSmafAct] = 0) : (D18F4x118/D18F4x11C[CpuPrbEnCstAct] = 0)) then this field specifies how long the core clock is left up to service additional probes before being brought back down. Each time a probe request is received, the hysteresis timer is reset such that the period of time specified by this field must expire with no probe request before the core clock is brought back down. The hysteresis time is encoded as (the time base specified by D18F3xD4[ClkRampHystCtl]) * (1 + ClkRampHystSel). • If (D18F4x128[CoreCstateMode] ? (D18F3x80/D18F3x84[CpuPrbEnSmafAct] = 1) : (D18F4x118/D18F4x11C[CpuPrbEnCstAct] = 1)) then this field specifies a fixed amount of time to allow for probes to be serviced after completing the transition of each core. If, for example, two cores enter stop-grant or halt at the same time, then (1) the first core would complete the transition to the low power state, (2) probe traffic would be serviced for the time specified by this field, (3) the second core would complete the transition to the low power state, and (4) probe traffic would be seviced for the time specified by this field (and afterwards, until the next power state transition). For this purpose, values range from 0h=40 ns to Fh=640 ns, encoded as 40 ns * (1 + ClkRampHystSel).
7:0	Reserved.

D18F3xD8 Clock Power/Timing Control 1

See 2.5.1.4 [Voltage Transitions].

Bits	Description
31:7	Reserved.
6:4	**VSRampSlamTime**. Read-write. Cold reset: 000b. BIOS: 100b. Specifies the time the processor waits for voltage transitions to complete before beginning an additional voltage change or a frequency change. Wait time = (VSRampSlamTime / 15mV) * ABS(destination voltage - current voltage). Bits / Description / Bits / Description 000b / 5.00 us / 100b / 2.00 us 001b / 3.75 us / 101b / 1.50 us 010b / 3.00 us / 110b / 1.20 us 011b / 2.40 us / 111b / 1.00 us
3:0	Reserved.

D18F3xDC Clock Power/Timing Control 2

Bits	Description
31:30	**NbsynPtrAdjPstate[2:1]: NB/core synchronization FIFO pointer adjust P-state[2:1]**. Read-write. Reset: Product-specific. See NbsynPtrAdj.
29:27	**NbsynPtrAdjLo: NB/core synchronization FIFO pointer adjust low**. Read-write. Cold reset: 000b. See NbsynPtrAdj.

26	**IgnCpuPrbEn: ignore CPU probe enable**. Read-write. Cold reset: 0. BIOS: 1. See D18F3x80/D18F3x84[CpuPrbEnSmafAct] and D18F4x118/D18F4x11C[CpuPrbEnCstAct].
25:19	**CacheFlushOnHaltTmr: cache flush on halt timer**. Read-write. Cold reset: 00h. BIOS: 1Bh. Specifies how long each core needs to stay in a C-state before it flushes its caches. See CacheFlushOnHaltCtl, D18F3x80/D18F3x84[CpuPrbEnSmafAct], and D18F4x118/D18F4x11C[CacheFlushTmrSel]. Bits: 00h — 5.12 us Bits: 7Fh-01h — (<CacheFlushOnHaltTmr> * 10.24 us) - 5.12 us <= Time <= <CacheFlushOnHaltTmr> * 10.24 us
18:16	**CacheFlushOnHaltCtl: cache flush on halt control**. Read-write. Reset: 000b. BIOS: 111b. Enables cache flush on halt when (CacheFlushOnHaltCtl != 0). Specifies what core clock divisor is used after the caches have been flushed. See D18F4x118/D18F4x11C[CacheFlushTmrSel]. Bits: 000b — Divide by 1 Bits: 001b — Divide by 2 Bits: 010b — Divide by 4 Bits: 011b — Divide by 8 Bits: 100b — Divide by 16 Bits: 101b — Reserved Bits: 110b — Reserved Bits: 111b — Turn off clocks See D18F3x[84:80] and D18F4x11[C:8] for clock divisor specifications that are in effect during a C-state before the caches have been flushed. See 2.5.2.2.3.1 [C-state Probes and Cache Flushing].
15	**NbsynPtrAdjPstate[0]: NB/core synchronization FIFO pointer adjust P-state[0]**. Read-write. Reset: Product-specific. See NbsynPtrAdj.

<table>
<tr><td>14:12</td><td>NbsynPtrAdj: NB/core synchronization FIFO pointer adjust. Read-write; Set-by-hardware. Cold reset: 000b.
This field is controlled by the APU hardware and software should not write to it.
Changes to this field take effect after any of the following events:
• Warm reset.
• At least one core on all compute units perform a P-state transition.
• An NB P-state transition.
There is a synchronization FIFO between the NB clock domain and core clock domains. At cold reset, the read pointer and write pointer for each of these FIFOs is positioned conservatively, such that FIFO latency may be greater than is necessary.

NbsynPtrAdj and NbsynPtrAdjLo may be used to position the read pointer and write pointer of each FIFO closer to each other such that latency is reduced. Each increment of NbsynPtrAdj and NbsynPtrAdjLo represents one clock cycle of whichever is the slower clock (longer period) between the NB clock and the core clock. NbsynPtrAdj is used when the core P-state is less than or equal to NbsynPtrAdjPstate, otherwise NbsynPtrAdjLo is used.

Values less than the recommended value are allowed; values greater than the recommended value are illegal.
<table><tr><td>Bits</td><td>Description</td></tr><tr><td>6h-0h</td><td>Position the read pointer <NbsynPtrAdj, NbsynPtrAdjLo> clock cycles closer to the write pointer.</td></tr><tr><td>7h</td><td>Reserved.</td></tr></table></td></tr>
<tr><td>11</td><td>Reserved.</td></tr>
<tr><td>10:8</td><td>HwPstateMaxVal: P-state maximum value. Read-write. IF ((D18F3xE8[HtcCapable] == 1) && (D0F0xBC_xD820_0C64[HTC_TMP_LMT] !=0) && (D0F0xBC_xD820_0C64[HTC_PSTATE_LIMIT] > HwPstateMaxVal)) THEN BIOS: D0F0xBC_xD820_0C64[HTC_PSTATE_LIMIT]. ENDIF. Cold reset: specified by the reset state of MSRC001_00[6B:64][PstateEn]; the cold reset value is the highest P-state number corresponding to the register in which PstateEn is set (e.g., if MSRC001_0064 and MSRC001_0065 have this bit set and the others do not, then HwPstateMaxVal = 1; if MSRC001_0064 has this bit set and the others do not, then HwPstateMaxVal = 0). This specifies the highest P-state value (lowest performance state) supported by the hardware. This field must not be written to a value less (higher performance) than MSRC001_0071[CurPstateLimit].
HwPstateMaxVal can only be written with values that are greater than or equal to D18F4x15C[NumBoostStates]. Writes to HwPstateMaxVal with values less than D18F4x15C[NumBoostStates] are ignored.
See MSRC001_0061[PstateMaxVal]. This field uses hardware P-state numbering. See 2.5.2.1.1.2 [Hardware P-state Numbering].</td></tr>
<tr><td>7</td><td>FastSprSaveRestEn. Read-write. Reset: 0. BIOS: 1. 1=Pack 16 SPRs per cache line when performing Save/Restore using the scrubber logic.</td></tr>
<tr><td>6:0</td><td>Reserved.</td></tr>
</table>

D18F3xE8 Northbridge Capabilities

Read-only. Value: Product-specific. Unless otherwise specified, 1=The feature is supported by the processor; 0=The feature is not supported.

Bits	Description
31:29	Reserved.
28	**SUCCOR**. Read-only. See CPUID Fn8000_0007_EBX[SUCCOR]. Value: 0.
27:25	Reserved.
24	**MemPstateCap: memory P-state capable**.
23:20	Reserved.
19	**x2Apic: x2APIC capability**. Value: 0.
18:15	Reserved.
14	**MultVidPlane: multiple VID plane capable**. Value: 1.
13:11	Reserved.
10	**HtcCapable: HTC capable**. This affects D0F0xBC_xD820_0C64 and D18F3x68.
9	**SvmCapable: SVM capable**.
8	**MctCap: memory controller (on the processor) capable**. Value: 1.
7:5	Reserved.
4	**ChipKill: chipkill ECC capable**.
3	**ECC: ECC capable**.
2	**EightNode: Eight-node multi-processor capable**.
1	**DualNode: Dual-node multi-processor capable**.
0	Reserved.

D18F3xF0 DEV Capability Header Register

The DEV secure loader function is configured through D18F3xF4 and D18F3xF8. The register number (i.e., the number that follows F8_x in the register mnemonic) is specified by D18F3xF4[DevFunction]. Access to this register is accomplished as follows:

- Reads: Write the register number to D18F3xF4[DevFunction]. Read the register contents from D18F3xF8.
- Writes: Write the register number to D18F3xF4[DevFunction]. Write the register contents to D18F3xF8.

IF (D18F3xE8[SvmCapable] == 0) THEN

Bits	Description
31:0	Reserved.

ELSE

Bits	Description
31:0	Reserved.

ENDIF.

D18F3xF4 DEV Function Register

Reset: 0000_0000h. Reserved if (D18F3xE8[SvmCapable] == 0).

Bits	Description
31:0	Reserved.

D18F3xF8 DEV Data Port

Reset: 0000_0000h. Address: D18F3xF4[DevFunction]. See D18F3xF0 for details about this port.

Bits	Description
31:0	Reserved.

D18F3xF8_x4 DEV Secure Loader Control Register

Reset: 0000_0000h.

Bits	Description
31:8	Reserved.
7:5	Reserved.
4:0	Reserved.

D18F3xFC CPUID Family/Model/Stepping

CPUID Fn0000_0001_EAX, CPUID Fn8000_0001_EAX are an alias of D18F3xFC.

Bits	Description
31:28	Reserved.
27:20	**ExtFamily: extended family**. Read-only. Value: 06h.
19:16	**ExtModel: extended model**. Read-only. Value: Product-specific.
15:12	Reserved.
11:8	**BaseFamily**. Read-only. Reset: Fh.
7:4	**BaseModel**. Read-only. Value: Product-specific.
3:0	**Stepping**. Read-only. Value: Product-specific.

D18F3x140 SRI to XCS Token Count

Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

D18F3x140, D18F3x144, and D18F3x1[54,50,4C,48] specify the number of XCS (XBAR command scheduler) entries assigned to each virtual channel within each source port. See 2.8 [Northbridge (NB)]. The default totals are:

Buffer allocation rules:

- SUM(D18F3x140[UpReqTok, UpPreqTok, UpRspTok, DnReqTok, DnPreqTok, DnRspTok, IsocReqTok, IsocPreqTok, IsocRspTok, FreeTok]) + SUM(D18F3x144[ProbeTok, RspTok]) + SUM(D18F3x148[ReqTok0, PReqTok0, RspTok0, ProbeTok0, {FreeTok[3:2],FreeTok[1:0]}, IsocReqTok0, IsocPReqTok0, IsocRspTok0, ReqTok1, PReqTok1, RspTok1, ProbeTok1, IsocReqTok1, IsocPReqTok1, IsocRspTok1]) + SUM(D18F3x14C[ReqTok0, PReqTok0, RspTok0, ProbeTok0, {FreeTok[3:2],FreeTok[1:0]}, IsocReqTok0, IsocPReqTok0, IsocRspTok0, ReqTok1, PReqTok1, RspTok1, ProbeTok1, IsocReqTok1, IsocPReqTok1, IsocRspTok1]) <= XcsSize. See D18F3x1[54,50,4C,48].

The defaults for D18F3x140 and D18F3x1[54,50,4C,48]do not allocate any tokens in the isochronous channel. If isochronous flow control mode (IFCM) is enabled (D18F0x[E4,C4,A4,84][IsocEn]), then the XCS token counts must be changed.

- If IFCM is enabled, then D18F3x140[IsocReqTok, IsocRspTok] must each be non-zero. If isochronous posted requests may be generated in the system, then D18F3x140[IsocPreqTok] must also be non-zero.
- If an IOMMU is present, D18F3x1[54,50,4C,48][IsocReqTok] must be non-zero.

Bits	Description
31:25	Reserved.
24:20	**FreeTok: free tokens**. Read-write. Reset: 08h. BIOS: Dh. The number of free tokens must always be greater than or equal to 2 to ensure deadlock free operation.
19:18	Reserved.
17:16	**IsocRspTok: isochronous response tokens**. Read-write. Reset: 0. BIOS: 1.
15:14	**IsocPreqTok: isochronous posted request tokens**. Read-write. Reset: 0.
13:12	**IsocReqTok: isochronous request tokens**. Read-write. Reset: 0. BIOS: 1.
11:10	**DnRspTok: downstream response tokens**. Read-write. Reset: 1.
9:8	**UpRspTok: upstream response tokens**. Read-write. Reset: 3. BIOS: 1.
7:6	**DnPreqTok: downstream posted request tokens**. Read-write. Reset: 1.
5:4	**UpPreqTok: upstream posted request tokens**. Read-write. Reset: 1. BIOS: 2.
3:2	**DnReqTok: downstream request tokens**. Read-write. Reset: 1.
1:0	**UpReqTok: upstream request tokens**. Read-write. Reset: 3. BIOS: 2.

D18F3x144 MCT to XCS Token Count

See D18F3x140.

Bits	Description
31:9	Reserved.
8:4	**RspTok: response tokens**. Read-write. Reset: 7h.
3:0	**ProbeTok: probe tokens**. Read-write. Reset: 7h. BIOS: 4h.

D18F3x1[54,50,4C,48] Link to XCS Token Count

See D18F3x140.

Table 193: Register Mapping for D18F3x1[54,50,4C,48]

Register	Function
D18F3x148	ONION Link
D18F3x1[54:4C]	Reserved

Bits	Description
31:30	**FreeTok[3:2]: free tokens**. Read-write. Cold reset: 00b. BIOS: 0. See FreeTok[1:0].
29	Reserved.
28	**IsocRspTok1: isochronous response tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
27	Reserved.
26	**IsocPreqTok1: isochronous posted request tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
25	Reserved.
24	**IsocReqTok1: isochronous request tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
23:22	**ProbeTok1: probe tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
21:20	**RspTok1: response tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
19:18	**PReqTok1: posted request tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
17:16	**ReqTok1: request tokens sublink 1**. Read-write. Cold reset: 0. BIOS: 0.
15:14	**FreeTok[1:0]: free tokens**. Read-write. Cold reset: 00b. FreeTok[3:0] = {FreeTok[3:2], FreeTok[1:0]}. BIOS: 10b.
13:12	**IsocRspTok0: isochronous response tokens sublink 0**. Read-write. Cold reset: 0. BIOS: 0.
11:10	**IsocPreqTok0: isochronous posted request tokens sublink 0**. Read-write. Cold reset: 0. BIOS: IF (REG == D18F3x148) THEN 1. ELSE 0. ENDIF. See D18F0x6C[ApplyIsocModeEnNow].
9:8	**IsocReqTok0: isochronous request tokens sublink 0**. Read-write. Cold reset: 0. BIOS: IF (REG == D18F3x148) THEN 1. ELSE 0. ENDIF.
7:6	**ProbeTok0: probe tokens sublink 0**. Read-write. Cold reset: 2. BIOS: 0.
5:4	**RspTok0: response tokens sublink 0**. Read-write. Cold reset: 2. BIOS: IF (REG == D18F3x148) THEN 2. ELSE 0. ENDIF.
3:2	**PReqTok0: posted request tokens sublink 0**. Read-write. Cold reset: 2. BIOS: IF (REG == D18F3x148) THEN 2. ELSE 0. ENDIF.
1:0	**ReqTok0: request tokens sublink 0**. Read-write. Cold reset: 2. BIOS: IF (REG == D18F3x148) THEN 2. ELSE 3. ENDIF.

D18F3x160 NB Machine Check Misc (DRAM Thresholding) 0 (MC4_MISC0)

Bits	Description
31:0	Reserved.

D18F3x168 NB Machine Check Misc (Link Thresholding) 1 (MC4_MISC1)

See 2.16.1.7 [Error Thresholding]. D18F3x168 is associated with the link error type. See MSRC000_0408.

Bits	Description
31	**Valid**. Read-only. Reset: 1.
30	**CntP: counter present**. Read-only. Reset: 1.
29	**Locked**. Read-only. Reset: 0.
28	**IntP**. Read-only. Reset: 1.
27:24	Reserved.
23:20	**LvtOffset: LVT offset**. IF (Locked) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. BIOS: 1.
19	**CntEn: counter enable**. IF (Locked) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
18:17	**IntType: interrupt type**. IF (Locked) THEN Read-only. ELSE Read-write. ENDIF. Cold reset: 0.
16	**Ovrflw: overflow.** IF (Locked) THEN Read-only; Set-by-hardware. ELSE Read-write; Set-by-hardware. ENDIF. Cold reset: 0.
15:12	Reserved.
11:0	**ErrCnt: error counter**. IF (Locked) THEN Read-only; Updated-by-hardware. ELSE Read-write; Updated-by-hardware. ENDIF. Cold reset: 0.

D18F3x170 NB Machine Check Misc (L3 Thresholding) 2 (MC4_MISC2)

Bits	Description
31:0	Reserved.

D18F3x17C Extended Freelist Buffer Count

Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

Bits	Description
31:11	Reserved.
10:6	**XbarToSrqFreeListBufCnt: XBAR to SRQ free list command buffer**. Read-write. Reset: 01h. BIOS: 02h. Determines the number of freelist buffers which can be allocated from the SRQ freelist pool to the XBAR.
5	Reserved.
4:0	**SPQPrbFreeCBC: XBAR to SRI Probe command buffer freelist**. Read-write. Reset: 08h. BIOS: 0Ch.

D18F3x180 Extended NB MCA Configuration

Reset: 0000_0000h. This register is an extension of D18F3x44 [MCA NB Configuration].
See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture.

Bits	Description
31:30	Reserved.
29	**SyncFloodOnDramUncCrcErr**. Read-write. 1=Enable generation of Sync flood on DRAM Uncorrectable CRC Error.
28:27	Reserved.
26	**ConvertUnCorToCorErrEn: convert uncorrectable error to correctable error enable**. Read-write. 1=The status of uncorrectable errors is changed to appear as correctable errors; D18F3x4C[UC, PCC] are cleared and a machine check exception is not raised. For uncorrectable ECC errors, D18F3x4C[UECC] is cleared and D18F3x4C[CECC] is set. This field is intended for debug observability.
25	Reserved.
24	**McaLogErrAddrWdtErr: log error address on WDT errors**. Read-write. BIOS: 1. 1=When a watchdog timeout error occurs (see D18F3x40[WDTRptEn]), the associated address is logged and D18F3x4C[AddrV] is set. 0=When a watchdog timeout error occurs, NB state information is saved and D18F3x4C[AddrV] is cleared. See D18F3x50 for details on saved information.
23	**SyncFloodOnDramTempErr**. Read-write. 1=Sync flood is generated on a DRAM temp Error.
22	Reserved.
21	**SyncFloodOnCpuLeakErr: sync flood on CPU leak error**. Read-write. BIOS: 1. 1=Enable Sync flood when one of the cores encounters an uncorrectable error which cannot be contained to the process on the core.
20	Reserved.
19	**PwP2pDatErrRmtPropDis: posted write for remote peer-to-peer data error propagation disable**. Read-write. 1=A peer-to-peer posted write with a data error is not propagated to the target IO link chain if the target IO link chain is not attached to the local node (the same node as the source IO link chain). Instead, the write is dropped by the host bridge. The state of this field is ignored if SyncFloodOnUsPwDatErr == 1 or DatWrErrDeferEn == 0.
18	**PwP2pDatErrLclPropDis: posted write for local peer-to-peer data error propagation disable**. Read-write. 1=A peer-to-peer posted write with a data error is not propagated to the target IO link chain if the target IO link chain is attached to the local node (the same node as the source IO link chain). Instead, the write is dropped by the host bridge. The state of this field is ignored if SyncFloodOnUsPwDatErr == 1 or DatWrErrDeferEn == 0.
17	**SyncFloodOnDeferErrToIO: convert deferred error for an IO link to sync flood enable**. Read-write. BIOS: 1. 1=A deferred error which targets an IO link device is turned into a Sync flood. • When DramErrDeferEn is set and the read response is for a DMA read with a data error, setting SyncFloodOnDeferErrToIO causes a Sync flood. • When DatWrErrDeferEn is set and the write is peer-to-peer, setting SyncFloodOnDeferErrToIO causes a Sync flood.

16	**DeferDatErrNcHtMcaEn: convert deferred error for an IO link to machine check exception enable**. IF (D18F3xE8[SUCCOR]) THEN Read-write. ELSE Read-only. ENDIF. 1=A deferred error which targets an IO link device is turned into a machine check exception. • When DramErrDeferEn is set and the read response is for a DMA read with a data error, setting DeferDatErrNcHtMcaEn causes an uncorrected error to be logged and a machine check exception to be generated. An error response is returned to the IO device irrespective of the setting of DeferDatErrNcHtMcaEn. • When DatWrErrDeferEn is set and the write is peer-to-peer, setting DeferDatErrNcHtMcaEn causes an uncorrected error to be logged and a machine check exception to be generated. An error indication is sent to the target IO device irrespective of the setting of DeferDatErrNcHtMcaEn.
15:13	Reserved.
12	**DramErrDeferEn: enable deferred errors on uncorrectable DRAM ECC errors**. IF (D18F3xE8[SUCCOR]) THEN Read-write. ELSE Read-only. ENDIF.
11	Reserved.
10	Reserved.
9	**SyncFloodOnUCNbAry: sync flood on UC NB array error**. Read-write. BIOS: 1. 1=Enable Sync flood on detection of an UC error in an NB array.
8	**SyncFloodOnProtErr: sync flood on protocol error**. Read-write. BIOS: 1. 1=Enable Sync flood on detection of link protocol error, L3 protocol error, and probe filter protocol error.
7	**SyncFloodOnTgtAbortErr**. Read-write. BIOS: 1. 1=Enable Sync flood on generated or received link responses that indicate target aborts.
6	**SyncFloodOnDatErr**. Read-write. BIOS: 1. 1=Enable Sync flood on generated or received link responses that indicate data error.
5	**DisPciCfgCpuMstAbortRsp**. Read-write. BIOS: 1. 1=For master abort responses to CPU-initiated configuration accesses, disables MCA error reporting and generation of an error response to the core. It is recommended that this bit be set in order to avoid MCA exceptions being generated from master aborts for PCI configuration accesses, which are common during device enumeration.
4	**ChgMstAbortToNoErr**. Read-write. 1=Signal no errors instead of master abort in link response packets to IO devices on detection of a master abort condition. When ChgMstAbortToNoErr and D18F3x44[IoMstAbortDis] are both set, ChgMstAbortToNoErr takes precedence.
3	**ChgDatErrToTgtAbort**. Read-write. 1=Signal target abort instead of data error in link response packets to IO devices (for Gen1 link compatibility).
2	**WDTCntSel[3]: watchdog timer count select bit[3]**. Read-write. See D18F3x44[WDTCntSel].
1	**SyncFloodOnUsPwDatErr: sync flood on upstream posted write data error**. Read-write. BIOS: 1. 1=Enable Sync flood generation when an upstream posted write data error is detected.
0	**McaLogUsPwDatErrEn: MCA log of upstream posted write data error enable**. Read-write. BIOS: 1. 1=Enable logging of upstream posted write data errors in MCA (if NB MCA registers are appropriately enabled and configured).

D18F3x188 NB Configuration 2

Same-for-all.

Bits	Description
31:28	Reserved.

27	**DisCpuWrSzDw64ReOrd: disable streaming store reorder**. Read-write. Reset: 1. BIOS: 1. 1=Disable reordering of streaming store commands.
26:10	Reserved.
9	**DisL3HiPriFreeListAlloc**. Read-write. Reset: 0. BIOS: 1. 1=Disables normal SRQ entry scheme which gives higher priority to XBAR.
8:0	Reserved.

D18F3x190 Downcore Control

Cold reset: 0000_0000h. See 2.4.4 [Processor Cores and Downcoring] and 2.4.4.1 [Software Downcoring using D18F3x190[DisCore]].

Bits	Description
31:0	**DisCore**. Read-write; reset-applied. 0=Core enabled. 1=Core disabled. Bit — Description [0] — Core 0. [2:1] — Core <BIT>. [3] — Core 3. [31:4] — Reserved.

D18F3x198 MCA NB Control Mask (MC4_CTL_MASK)

IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Not-same-for-all. ELSE Read-write; Per-node; Not-same-for-all. ENDIF. The format of MC4_CTL_MASK corresponds to D18F3x40 [MCA NB Control]. For each defined bit position, 1=Disable logging. MCi_CTL_MASK allow BIOS to mask the presence of any error source from software for test and debug. When error sources are masked, it is as if the error was not detected. Such masking consequently prevents error responses and actions. Only one of these registers exists in multi-core devices. See 2.16.1 [Machine Check Architecture]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn].

Bits	Description
31	Reserved.
30	Reserved.
29	**WrCrcErr**. Reset: 0.
28	**G5CrcCorrErr**. Reset: 0.
27	Reserved.
26	**NbArrayPar**. Reset: 1. IF (D18F4x118[NbPwrGate0]==1 \|\| D18F4x118[NbPwrGate1] == 1 \|\| D18F4x11C[NbPwrGate2] == 1) BIOS: 1. ELSE BIOS: 0. ENDIF.
25	**McaUsPwDatErr**. Reset: 0.
24	Reserved.
23	**CrcErr[3]**. Reset: 0. See CrcErrEn[2:0].

22:19	**RtryHt**. Reset: 0. BIOS is recommended to mask logging HyperTransport™ retries if the OS is not capable of distinguishing that HyperTransport retries are normal operation. Bit Description [3] Link 3 [2] Link 2 [1] Link 1 [0] Link 0
18	**DramPar**. Reset: 0.
17	Reserved.
16	**Prot**. Reset: 0.
15	**NBArrayUnc**. Reset: 0.
14	**NBArrayCor**. Reset: 0.
13	**DevErr**. Reset: 0.
12	**WDTRpt**. Reset: 0.
11	**AtomicRMW**. Reset: 0.
10	**GartTblWk**. Reset: 0.
9	Reserved.
8	Reserved.
7:5	**SyncPkt[2:0]**. Reset: 0. SyncPkt[3:0] = {SyncPkt[3], SyncPkt[2:0]}. Bit Description [3] Link 3 [2] Link 2 [1] Link 1 [0] Link 0
4:2	**CrcErr[2:0]**. Reset: 0. CrcErr[3:0] = {CrcErr[3], CrcErr[2:0]}. Bit Description [3] Link 3 [2] Link 2 [1] Link 1 [0] Link 0
1	**UnCorrEcc**. Reset: 0.
0	**CorrEcc**. Reset: 0.

D18F3x1A0 Core Interface Buffer Count

Out of cold reset, the processor allocates a minimal number of buffers that is smaller than the default values in the register. BIOS must use D18F0x6C[RlsLnkFullTokCntImm] for the values in the register to take effect. This is necessary even if the values are unchanged from the default values.

- The following buffer allocations rules must be satisfied:
 - CpuCmdBufCnt >= 2.

Bits	Description
31	Reserved.

30:26	**NbToCpuPrbLmt**. Read-write. Reset: 0Fh. Maximum number of outstanding probes to the compute-unit. Bits Description 02h-00h Reserved. 0Fh-03h Maximum of <NbToCpuPrbLmt> probes. 1Fh-10h Reserved.
25:24	Reserved.
23:20	**NbToCpuDatReqLmt**. Read-write. Reset: Ch. Octoword outstanding per core limit. Bits Description Ch-0h Octoword outstanding per core limit. Fh-Dh Reserved.
19	Reserved.
18:16	**CpuToNbFreeBufCnt**. Read-write. Reset: 2h. BIOS: 5h. Provides the number of tokens which can released to each compute unit from the freelist pool. This field can be updated at any time by BIOS and does not require a warm reset to take effect.
15:12	Reserved. Reset: 4h.
11:3	Reserved.
2:0	**CpuCmdBufCnt: CPU to SRI command buffer count**. Read-write; reset-applied. Reset: 2h. BIOS: 1h. Each compute unit is allocated the number of buffers specified by this field.

D18F3x1CC IBS Control

Reset: 0000_0000h. MSRC001_103A is an alias of D18F3x1CC. D18F3x1CC is programmed by BIOS; The OS reads the LVT offset from MSRC001_103A.

Bits	Description
31:9	Reserved.
8	**LvtOffsetVal: local vector table offset valid**. Read-write. BIOS: 1. 1=The offset in LvtOffset is valid. 0=The offset in LvtOffset is not valid and IBS interrupt generation is disabled.
7:4	Reserved.
3:0	**LvtOffset: local vector table offset**. Read-write. BIOS: 0. Specifies the address of the IBS LVT entry in the APIC registers. See APIC[530:500]. Bits Description 3h-0h LVT address = <500h + LvtOffset<<4> Fh-4h Reserved

D18F3x200 Performance Mode Control Register

Bits	Description
31:27	Reserved.
26	**EnSpqUseSrqFree**. Read-write. Reset: 1. 1=Enable Probe repsonses to be able to use the SRQ freelist buffer pool. There are 4 dedicated SRQ tokens for probe responses.
25	**EnWriteBackChan**. Read-write. Reset: 0. 1=Enable Write-Back channel.

24:19	**CpuFlbThrottleCnt**. Read-write. Reset: 08h. Lower limt for number of CPU writes per CPS used to trhottle CPU writes to SRQ entries. Bits Description 0Fh-00h Minimum CPU Writes 3Fh-10h Reserved
18	**EnSetCpuDstValidOnAlloc**. Read-write. Reset: 1. 1=Destination valid bit set during the allocation of CPU requests accessing local DRAM memory. This helps in reducing the writes cancelled due to ordering in SP6 and hence, helps with the SRQ picker bandwidth.
17:15	**CpuWrPickThrottleCnt**. Read-write. Reset: 2. Programmed value determines the number of outstanding writes in the SRQ pipe. Enabled by D18F3x200[EnCpuWrPickThrottle]. Bits Description Bits Description 000b Reserved. 100b Reserved. 001b 1 101b Reserved. 010b 2 110b Reserved. 011b 3 111b Reserved.
14	**EnCpuWrPickThrottle**. Read-write. Reset: 1. 1=Enables the write pick throttling based on CpuWrPickThrottleCnt in SRQ.
13:8	**CpuWrReqThrottleCnt**. Read-write. Reset: 10h. Maximum number of outstanding writes per CPC in SRQ. Bits Description 00h Reserved 3Fh-01h Maximum Outstanding Writes
7:4	**EnCpuSkidBufFull**. Read-write. Reset: 1. Enables optimal use of the CPU skid buffers, in the presence of multiple data movement requests from the same core.
3	**EnMcqPrbPickThrottle**. Read-write. Reset: 0. BIOS: 1. 1=Enabling throttling the MCQ to ensure the bypass path is taken by the probes instead of allocating in to the XCS.
2	**EnDctOddToNcLnkDatXfr**. Read-write. Reset: 1. BIOS: 1. 1=Enables direct transfer of Read Response Data from odd-numbered DRAM channels (1, 3, ...) to non-coherent links on the local node.
1	**EnDctEvnToNcLnkDatXfr**. Read-write. Reset: 1. 1=Enables direct transfer of data from even-numbered DRAM channels (0, 2, ...) to non-coherent links on the local node.
0	Reserved.

D18F3x240 MCT Configuration Low (MCT_CFG2_LO)

Bits	Description
31:29	**McqNonDramPriByPassMax**. Read-write. Reset: 100b. Number of non-DRAM priority access to allow before allowing a low priority non-DRAM access operation to occur. Setting to 000b is reserved.
28:25	**WrBurstWatchDogCntSel**. Read-write. Reset: 0.
24:20	Reserved.
19:14	**FavorXbarOverScrubsThreshold**. Read-write. Reset: 0.

13:12	**WcBufCloseTimeoutThreshold**. Read-write. Reset: 0. Defines the Age threshold in terms of NCLK cycles Bits: Description 00b: 7 NCLK cycles 01b: 15 NCLK cycles 10b: 23 NCLK cycles 11b: 31 NCLK Cycles
11	**WcBufCloseTimeoutEn**. Read-write. Reset: 1. 1=Enables WCB entries allocated by Onion3 writes to close out optionally using an age based timer.
10	**WcBufCloseBurstLmt**. Read-write. Reset: 0. 1=Close WCBs when the associated MCT Write Burst Threshold is asserted.
9:7	**WcBufEvicThreshold**. Read-write. Reset: 011b. Determines the threshold for number of eligible for WCB allocation that are dropped because all 4 WCBs are busy when WCBufReplaceEn == 1.
6	**WCBufReplaceEn**. Read-write. Reset: 0. Round-Robin WCB replacement. 1=Enable the counst of the number of eligible writes for WCB allocation that are dropped because all 4 WCBs are busy and it crosses a threshhold determined by WcBufEvicThreshold, closing a WCB.
5:4	**WcBufHitThresholdOn3**. Read-write. Reset: 01b. Determines the number of Hits a WCB entry can have for an Onion3 request before it closes. Bits: Description 00b: Reserved 01b: 1 Hit 10b: Reserved 11b: Reserved
3:2	**WcBufHitThreshold**. Read-write. Reset: 01b. Determines the number of Hits a WCB entry can have for a CPU/IO request before it closes. Bits: Description 00b: Unlimited 01b: 1 Hit 10b: 3 Hits 11b: 7 Hits
1:0	**WcBufAllocation**. Read-write. Reset: 00b. MCT WCB allocation mode. Bits: Description 00b: First come, first served (open to both CPU/IO and Onion3) 01b: All WCBs for CPU/IO 10b: All WCBs for Onion3 11b: Reserved

D18F3x244 MCT Configuration High (MCT_CFG2_High)

Bits	Description
31:30	Reserved.
29:24	**MctOn3CacheHitPredFallThrLnk1**. Read-write. Reset: 1Fh. Define the Falling Threshold for the Onion3 Link 1 counter. This is to allow for hysteresis. The Rising Threshold >= Falling Threshold. Extreme values (depending on the width of the window) for either Rising or Falling Thresholds are illegal.

23:18	**MctOn3CacheHitPredRiseThrLnk1**. Read-write. Reset: 1Fh. Define the Rising Threshold for the Onion3 Link 1 counter. This is to allow for hysteresis. The Rising Threshold >= Falling Threshold. Extreme values (depending on the width of the window) for either Rising or Falling Thresholds are illegal.
17:12	**MctOn3CacheHitPredFallThrLnk0**. Read-write. Reset: 1Fh. Define the Falling Threshold for the Onion3 Link 0 counter. This is to allow for hysteresis. The Rising Threshold >= Falling Threshold. Extreme values (depending on the width of the window) for either Rising or Falling Thresholds are illegal.
11:6	**MctOn3CacheHitPredRiseThrLnk0**. Read-write. Reset: 1Fh. Define the Rising Threshold for the Onion3 Link 0 counter. This is to allow for hysteresis. The Rising Threshold >= Falling Threshold. Extreme values (depending on the width of the window) for either Rising or Falling Thresholds are illegal.
5:4	**MctOn3CacheHitPredWindowLnk1**. Read-write. Reset: 11b. Sets the width of the window for probe responses for Onion3 link 1 coherent reads. Bits Description 00b 16 01b 32 10b 48 11b 64
3:2	**MctOn3CacheHitPredWindowLnk0**. Read-write. Reset: 11b. Sets the width of the window for probe responses for Onion3 link 0 coherent reads. Bits Description 00b 16 01b 32 10b 48 11b 64
1	**MctOn3CacheHitPredUnified**. Read-write. Reset: 0. 0=Per link counters look at probe responses for Onion3coherent requests from the respective links. 1=Link counters should look at probe responses for Onion3 coherent read requets from either link.
0	**DisMctOn3CacheHitPred**. Read-write. Reset: 0. 0=Enables cache hit prediction based on a temporal predictor per link, which looks at a window of the last N (N = 16, 32, 48 or 64) probe responses for Onion3 coherent reads and maintains a counter. There are programmable rising and falling thresholds to predict a cache hit or miss. On a cache hit prediction, the DRAM read request is held until the probe response comes back. If a prediction is a success, the DRAM read is simply killed. If a prediction fails, a misprediction penalty is incurred, resulting in serializing the DRAM read round trip latency with the Probe Response round trip latency.

D18F3x2A0 Core Interface Buffer Count Register 2

Bits	Description
31:11	Reserved.
10:6	**CpuL3PrbqStpGntLmt**. Read-write. Reset: Fh. Specifies the CPU probe limit when the NB is in StopGrant Mode.

5:4	Reserved.
3:0	**CpuToNbFreeBufCntHi**. Read-write. Reset: 0. BIOS: 5h. Specifies the maximum number of freelist tokens that may be released to a core. D18F3x1A0[CpuToNbFreeBufCnt] specifies the low watermark, requiring the CPU to return tokens above this watermark back to the NB in case it does not use them.

D18F3x2B4 DCT and Fuse Power Gate Control

Bits	Description
31:27	Reserved.
26	**FusePwrStatus**. Read-only. Cold reset: 0. Specifies whether fuses are power-gated. 1=Fuses are powered. 0=Fuses are powered-down.
25:24	**PwrDnHyst**. Read-write. Cold reset: 00b. Specifies the minimum amount of time that must expire with no fuse power up events before fuse power gating is initiated. Bits Description 00b 10 us 01b 32 us 10b 128 us 11b Reserved.
23:22	**PostPwrDnDelay**. Read-write. Cold reset: 00b. Specifies the amount of time between the completion of fuse power gating and the start of a new fuse power operation. Bits Description 00b 1 RefClk 01b 64 RefClks 10b 128 RefClks 11b 256 RefClks
21:20	Reserved.
19:18	**PostPwrUpDelay**. Read-write. Cold reset: 00b. Specifies the amount of time between the completion of fuse power ungating and the start of a new fuse power operation. Bits Description 00b 1 RefClk 01b Reserved. 10b Reserved. 11b Reserved.
17:16	**PrePwrUpDelay**. Read-write. Cold reset: 00b. Specifies the amount of time between a power up event and the start of fuse power ungating. Bits Description 00b 1 RefClk 01b Reserved. 10b Reserved. 11b Reserved.

15:10	Reserved.
9:8	**DctClkGateEn**. Read-write. Cold reset: 00b. BIOS: 0. 1=Enable DCT clock gating. 0=Disable DCT clock gating. [0]=DCT 0; [1]=DCT 1; [3:2]=Reserved. Once clock gating has been enabled, it cannot be disabled without a cold reset.
7:2	Reserved.
1:0	**DctPwrGateEn**. Read-write. Reset: 0. BIOS: 0. 1=Enable static DCT power gating. 0=Disable static DCT power gating. [0]=DCT 0; [1]=DCT 1.

D18F3x2E0 ONION3 Error Status

Reset: 0000_0000h. Read-write.

Bits	Description
31	**Valid: error valid**.
30	**Overflow: error overflow**.
29	**AddrVal: address valid**.
28:8	Reserved.
7:0	**ErrCode: error code**. Bits — Description [7:5] Destination; Legal values: 5h=Configuration, 1h=DRAM. [4:2] Source; Legal values: 4h=ONION3 Link0, 5h=ONION3 Link 1. [1:0] Access Type; 3h=Atomic, 2h=Write, 1h=Read, 0h=Other.

D18F3x2E4 ONION3 Error Address Low

Reset: 0000_0000h.

Bits	Description
31:2	**Onion3ErrAddr[31:2]: onion 3 error address low**. Read-write. See: D18F3x2E8[Onion3ErrAddr[47:32]].
1:0	Reserved.

D18F3x2E8 ONION3 Error Address High

Reset: 0000_0000h.

Bits	Description
31:16	Reserved.
15:0	**Onion3ErrAddr[47:32]: onion 3 error address high**. Read-write. Onion3ErrAddr[47:2] = {D18F3x2E8[Onion3ErrAddr[47:32]], D18F3x2E4[Onion3ErrAddr[31:2]]}.

3.14 Device 18h Function 4 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D18F4x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 15B4h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D18F4x04 Status/Command

Bits	Description
31:16	**Status**. Read-only. Reset: 0000_0000_000X_0000b. Only Status[4] may be set to indicate the existence of a PCI-defined capability block. 0=No supported links are unganged. 1=At least one link may be unganged, in which case there is a capability block associated with sublink one of the link in this function.
15:0	**Command**. Read-only. Value: 0000h.

D18F4x08 Class Code/Revision ID

Reset: 0600_0000h.

Bits	Description
31:8	**ClassCode**. Read-only. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only.

D18F4x0C Header Type

Reset: 0080_0000h.

Bits	Description
31:0	**HeaderTypeReg**. Read-only. These bits are fixed at their default values. The header type field indicates that there are multiple functions present in this device.

D18F4x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Value: 00h.

D18F4x110 Sample and Residency Timers

Bits	Description
31:21	Reserved.
20:13	**MinResTmr: minimum residency timer**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Cold reset: Product-specific. Specifies the minimum amount of time required between TDP-initiated P-state transitions. The minimum amount of time is defined as MinResTmr * CSampleTimer * 5.12 us.
12	Reserved.
11:0	**CSampleTimer**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. Cold reset: 0. BIOS: 2h. Specifies the value that the internal CSampleTimer counter must increment to before expiring. When the internal CSampleTimer counter expires, it is reset to 0. See 2.5.8 [Application Power Management (APM)].

D18F4x11[C:8] C-state Control

D18F4x11[C:8] consist of three identical 16-bit registers, one for each C-state Action Field (CAF) associated with an IO address that is read to enter C-states. Refer to 2.5.2.2 [Core C-states].

- D18F4x118[15:0] specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr].
- D18F4x118[31:16] specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr]+1.
- D18F4x11C[15:0] specifies the actions attempted by the core when software reads from the IO address specified by MSRC001_0073[CstateAddr]+2.

D18F4x118 C-state Control 1

Bits	Description
31:30	Reserved.
29	**SelfRefrEarly1**. Read-write. Reset: 0. See: SelfRefrEarly0. IF (D18F4x128[CoreCstateMode] == 0) THEN 0.
28	**SelfRefr1**. Read-write. Reset: 0. See: SelfRefr0. BIOS: 1.
27	**NbClkGate1**. Read-write. Reset: 0. See: NbClkGate0. BIOS: 1.
26	**NbPwrGate1**. Read-write. Reset: 0. See: NbPwrGate0. BIOS: 1.
25	**PwrOffEnCstAct1**. Read-write; Updated-by-SMU. Reset: 0. See: PwrOffEnCstAct0. BIOS: 1.
24	**PwrGateEnCstAct1**. Read-write. Reset: 0. See: PwrGateEnCstAct0. BIOS: 1.
23:21	**ClkDivisorCstAct1**. Read-write. Reset: 0. See: ClkDivisorCstAct0. BIOS: 000b.
20	Reserved.
19:18	**CacheFlushTmrSelCstAct1**. Read-write. Reset: 0. See: CacheFlushTmrSelCstAct0. BIOS: 01b.
17	**CacheFlushEnCstAct1**. Read-write. Reset: 0. See: CacheFlushEnCstAct0. BIOS: 1. IF ((D18F4x128[CoreCstateMode] == 0) && ((D18F4x118[SelfRefr1] \|\| D18F4x118[NbClkGate1]) == 1) THEN 1. ELSE 0. ENDIF.
16	**CpuPrbEnCstAct1**. Read-write. Reset: 0. See: CpuPrbEnCstAct0. BIOS: 1.

15:14	Reserved.
13	**SelfRefrEarly0: allow early self-refresh**. Read-write. Reset: 0. 1=Allow self-refresh while cores in PC1 or CC1 are waiting for the cache flush timer to expire. 0=Wait for cache flush timer to expire before allowing self-refresh. See 2.5.6.2 [DRAM Self-Refresh] and 2.5.2.2.3.1 [C-state Probes and Cache Flushing]. IF (D18F4x128[CoreCstateMode] == 0) THEN 0. ENDIF.
12	**SelfRefr0: self-refresh**. Read-write. Reset: 0. BIOS: 1. 1=Allow DRAM self-refresh while in NB C-states. 0=Prevent DRAM self-refresh while in NB C-states. NbClkGate0 must be equal to SelfRefr0. See 2.5.6.2 [DRAM Self-Refresh] and 2.5.3.2 [NB C-states].
11	**NbClkGate0: NB clock-gating**. Read-write. Reset: 0. BIOS: 1.1=Allow clock-gating of the NB. 0=Prevent clock-gating of the NB. NbClkGate0 must be equal to SelfRefr0. See 2.5.3.2 [NB C-states].
10	**NbPwrGate0: NB power-gating**. Read-write. Reset: 0. BIOS: 1. 1=Allow power-gating of the NB. 0=Prevent power-gating of the NB. NbPwrGate0 can only be programmed to 1 if NbClkGate0 and SelfRefr0 are programmed to 1. See 2.5.3.2 [NB C-states].
9	**PwrOffEnCstAct0: power off enable**. Read-write; Updated-by-SMU. Reset: 0. BIOS: 1. 1=Package power off enable. CacheFlushEnCstAct0 is required to be set if this bit is set. PwrGateEnCstAct0 is required to be set if this bit is set. See 2.5.2.2.3.4 [Package C6 (PC6) State].
8	**PwrGateEnCstAct0: power gate enable**. Read-write. Reset: 0. BIOS: 1. 1=Core power gating is enabled. CacheFlushEnCstAct0 is required to be set if this bit is set. See 2.5.2.2.3.3 [Core C6 (CC6) State].

<table>
<tr><td>7:5</td><td>ClkDivisorCstAct0: clock divisor. Read-write. Reset: 0. BIOS: 000b.
Specifies the core clock frequency while in the low-power state before the caches are flushed. This divisor is relative to the current FID frequency, or:
• 100 MHz * (10h + MSRC001_00[6B:64][CpuFid[5:0]]) of the current P-state specified by MSRC001_0063[CurPstate].
If MSRC001_00[6B:64][CpuDid] of the current P-state indicates a divisor that is deeper than specified by this field, then no frequency change is made when entering the low-power state associated with this register.
<table><tr><th>Bits</th><th>Description</th><th>Bits</th><th>Description</th></tr><tr><td>000b</td><td>/1</td><td>100b</td><td>/16</td></tr><tr><td>001b</td><td>/2</td><td>101b</td><td>Reserved</td></tr><tr><td>010b</td><td>/4</td><td>110b</td><td>Reserved</td></tr><tr><td>011b</td><td>/8</td><td>111b</td><td>Turn off clocks.</td></tr></table>See CacheFlushTmrSelCstAct0.</td></tr>
<tr><td>4</td><td>Reserved.</td></tr>
<tr><td>3:2</td><td>CacheFlushTmrSelCstAct0: cache flush timer select. Read-write. Reset: 00b. BIOS: 10b. Specifies the timer to use for cache flush.
<table><tr><th>Bits</th><th>Cache flush timer</th></tr><tr><td>00b</td><td>0 us</td></tr><tr><td>01b</td><td>D18F3xDC[CacheFlushOnHaltTmr]</td></tr><tr><td>10b</td><td>D18F4x128[CacheFlushTmr]</td></tr><tr><td>11b</td><td>Reserved</td></tr></table>Each compute unit has one timer that is shared by all cores within the compute-unit.
D18F3xDC[CacheFlushOnHaltCtl] specifies the core clock divisor to use after the caches are flushed. Writing values greater than 10b result in 10b. See CacheFlushEnCstAct0 and CpuPrbEnCstAct0.</td></tr>
<tr><td>1</td><td>CacheFlushEnCstAct0: cache flush enable. Read-write. Reset: 0. BIOS: 1. 1=Cache flush enable. The cache flush timer starts counting when the C-state is entered. See CacheFlushTmrSelCstAct0 and 2.5.2.2.3.1 [C-state Probes and Cache Flushing].
IF ((D18F4x128[CoreCstateMode] == 0) && ((D18F4x118[SelfRefr0] || D18F4x118[NbClkGate0]) == 1) THEN 1. ELSE 0. ENDIF.</td></tr>
<tr><td>0</td><td>CpuPrbEnCstAct0: core direct probe enable. Read-write. Reset: 0. BIOS: 1. Specifies how probes are handled while in the low-power state. 0=When the probe request comes into the NB, the core clock is brought up to the COF (based on the current P-state), all outstanding probes are completed, the core waits for a hysteresis time based on D18F3xD4[ClkRampHystSel], and then the core clock is brought down to the frequency specified by ClkDivisorCstAct0. 1=The core clock does not change frequency; the probe is handled at the frequency specified by ClkDivisorCstAct0; this may only be set if:
• ClkDivisorCstAct0 specifies a divide by 1, 2, 4, 8, or 16 and NbCof <= 3.2 GHz.
• ClkDivisorCstAct0 specifies a divide by 1, 2, 4, or 8 and NbCof >= 3.4 GHz.
This bit also specifies functionality of the timer used for cache flushing. See CacheFlushTmrSelCstAct0.
• If CpuPrbEnCstAct0 == 0 and D18F3xDC[IgnCpuPrbEn] == 0, only the time when the core is in a non-C0 state and has its clocks ramped up to service probes is counted.
• If CpuPrbEnCstAct0 == 1 or D18F3xDC[IgnCpuPrbEn] == 1, all of the time the core is in a non-C0 state is counted.</td></tr>
</table>

D18F4x11C C-state Control 2

Reset: 0000_0000h. Read-write.

Bits	Description
31:14	Reserved.
13	**SelfRefrEarly2**. See: D18F4x118[SelfRefrEarly0].
12	**SelfRefr2**. See: D18F4x118[SelfRefr0].
11	**NbClkGate2**. See: D18F4x118[NbClkGate0].
10	**NbPwrGate2**. Read-write. 1=Allow clock-gating of the NB. 0=Prevent clock-gating of the NB. NbClkGate2 must be equal to SelfRefr2. See 2.5.3.2 [NB C-states].
9	**PwrOffEnCstAct2**. See: D18F4x118[PwrOffEnCstAct0].
8	**PwrGateEnCstAct2**. See: D18F4x118[PwrGateEnCstAct0].
7:5	**ClkDivisorCstAct2**. See: D18F4x118[ClkDivisorCstAct0].
4	Reserved.
3:2	**CacheFlushTmrSelCstAct2**. See: D18F4x118[CacheFlushTmrSelCstAct0].
1	**CacheFlushEnCstAct2**. See: D18F4x118[CacheFlushEnCstAct0].
0	**CpuPrbEnCstAct2**. See: D18F4x118[CpuPrbEnCstAct0].

D18F4x124 C-state Interrupt Control

Bits	Description
31:0	Reserved.

D18F4x128 C-state Policy Control 1

Reset: 0080_0000h.

Bits	Description
31	**CstateMsgDis: C-state messaging disable**. Read-write. Specifies whether any messages are sent to the FCH when a core enters or exits a C-state. 0=Messages are sent. 1=Messages are not sent. See 2.5.2.2.4.1 [FCH Messaging].
30:25	Reserved.
24:23	**CacheFlushSucMonMispredictAct: cache flush success monitor mispredict action**. Read-write. BIOS: 01b. Specifies the cache flush success monitor decrement when non-C0 residency is shorter than duration specified by CacheFlushSucMonTmrSel. Bits — Description 00b — reset counter to zero 01b — decrement by 1 10b — decrement by 2 11b — decrement by 3

22:21	**CacheFlushSucMonTmrSel: cache flush success monitor timer select**. Read-write. BIOS: 00b. Specifies the non-C0 duration used to increment the cache flush success monitor. Bits: Duration 00b: Use cache flush timer specified by D18F4x11[C:8] 01b: D18F3xDC[CacheFlushOnHaltTmr] 10b: D18F4x128[CacheFlushTmr] 11b: Reserved
20:18	**CacheFlushSucMonThreshold: cache flush success monitor threshold**. Read-write. BIOS: 111b. Flush the caches immediately if cache flushing is enabled and the cache flush success monitor count == CacheFlushSucMonThreshold. A value of 0 disables the cache flush success monitor. See D18F4x118/D18F4x11C[CacheFlushEn].
17:12	Reserved.
11:5	**CacheFlushTmr: cache flush timer**. Read-write. BIOS: 1Bh. Specifies how long each core needs to stay in a C-state before it flushes its caches. See D18F4x118/D18F4x11C[CacheFlushTmrSel]. Bits: Description 00h: <= 5.12 us 7Fh-01h: (<CacheFlushTmr> * 10.24 us) - 5.12 us <= Time <= <CacheFlushTmr> * 10.24 us
4:2	**HaltCstateIndex**. Read-write. Specifies the IO-based C-state that is invoked by a HLT instruction.
1	**CoreCstatePolicy**. Read-write. BIOS: 0. Specifies how the processor arbitrates voltage and frequency when different non-C0 C-state requests are received on each core in a compute unit. 0=Transition both cores to the shallower C-state request. 1=Transition both cores to the deeper C-state request. For instance, if core 0 gets a request to go to C2 and core 1 gets a request to go to C1, hardware looks at the setting of CoreCstatePolicy. If CoreCstatePolicy is programmed to 0, the processor sends both cores to C1. If CoreCstatePolicy is programmed to 1, the processor sends both cores to C2. BIOS should program this field to the same value in all nodes of a multi-node processor.
0	Reserved.

D18F4x13C SMU P-state Control

Reset: 0000_0000h. Read-only; Updated-by-SMU.

Bits	Description
31:4	Reserved.
3:1	**SmuPstateLimit**. Specifies the highest-performance P-state (lowest value) allowed. SmuPstateLimit is always bounded by MSRC001_0061[PstateMaxVal]. This field uses hardware P-state numbering. See MSRC001_0071[CurPstateLimit] and 2.5.2.1.1.2 [Hardware P-state Numbering].
0	**SmuPstateLimitEn**.

D18F4x15C Core Performance Boost Control

Bits	Description
31	**BoostLock**. Read-only. Reset: Product-specific. Specifies whether the following registers are Read-write, Read-only, or have special requirements related to writability. See individual register definitions for details. • MSRC001_00[6B:64][CpuFid[5:0], CpuDid, CpuVid]. • D18F4x110[MinResTmr] • D18F4x15C[NumBoostStates]. • D18F4x16C[CstateCnt, CstateBoost]. • D18F4x250[NodeTdpLimit]. • D18F5xEC[LSCacThreshold, LSPstate, LSCpNum]
30:11	Reserved.
10	**LpmTrigger**. RAZ; Updated-by-SMU. Reset: 0. 1=Send LPML and LPMV to the compute unit power monitor.
9	**LpmOnVidNop**. If D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Cold-reset: 0. 1=LPMx commands are transmitted on P-state changes and VID changes. 0=LPMx commands are transmitted only on VID changes.
8	**CstatePowerEn: C-state power enable**. If D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: 0. BIOS: 1.
7	**ApmMasterEn: APM master enable**. If D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: 0. BIOS: IF(D18F4x15C[NumBoostStates] == 0) THEN 0. ELSE 1. ENDIF. 1=Enables the ability to turn on features associated with APM when used in conjunction with the individual feature enable bits. See 2.5.8 [Application Power Management (APM)].
6:5	Reserved.
4:2	**NumBoostStates: number of boosted states**. IF (D18F4x15C[BoostLock] \| ApmMasterEn \| D18F2x1B4[SmuCfgLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. Specifies the number of P-states that are considered boosted P-states. See 2.5.8 [Application Power Management (APM)].
1:0	**BoostSrc: boost source**. IF D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: 0. BIOS: 2.5.2.1.5. Specifies whether CPB is enabled or disabled. Bits — Description 00b — Boosting disabled 01b — Boosting enabled 10b — Reserved 11b — Reserved

D18F4x16C APM TDP Control

Bits	Description
31:15	Reserved.

14	**CacUpC1**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. 1=Cac interface is up on C1 (non XC6) state. 0=Cac interface is down and Cstate scalers are used in place of Cac reads.
13	**CstateCores**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. Specifies how CstateCnt determines Cstate boost conditions. Bit Description 0h CstateCnt specifies the number of compute units. 1h CstateCnt specifies the number of cores.
12	Reserved.
11:9	**CstateCnt: C-state count**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. Specifies the number of cores or compute units (see CstateCores) that must be in CC6 before a transition can occur to a boosted P-state that is higher performance than the P-state specified by CstateBoost. A value of 0 disables access to P-states above CstateBoost.
8:6	**CstateBoost**. Read-write. Reset: Product-specific. Specifies the P-state which requires the number of cores or compute units (see CstateCores) specified in CstateCnt to be in CC6 before a transition to a higher performance (lower numbered) boosted P-state is allowed. CstateBoost must be less than or equal to D18F4x15C[NumBoostStates] otherwise undefined behavior results. If D18F4x15C[BoostLock] == 1, CstateBoost can only be written with values that are greater than or equal to the reset value. Attempts to write values less than the reset value are ignored. A value of 0 indicates that the C-state boost feature is not supported. This field uses hardware P-state numbering. See 2.5.2.1.1.2 [Hardware P-state Numbering].
5	**ApmTdpLimitSts: APM TDP limit status**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. This bit is set by hardware when D18F5xE8[ApmTdpLimit] changes.
4	**ApmTdpLimitIntEn: APM TDP limit interrupt enable**. Read-write. Reset: 0. BIOS: 1. 1=Enables the generation of an interrupt using APIC330 of each core when D18F5xE8[ApmTdpLimit] changes.
3	**TdpLimitDis**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. BIOS: 1. 1=Disables TDP limit checking and allows the processor to transition to higher performance P-states.
2:0	Reserved.

D18F4x170 Interrupt Rate Monitor Control

Reset: 0000_0000h

Bits	Description
31:16	Reserved.
15:11	**DecrRate: interrupt rate monitor decrement rate**. Read-write. BIOS: 18h. Specifies the rate at which the interrupt counter is decremented. The rate is expresed as (DecrRate * 50 us).

10:8	**BurstLen: interrupt rate monitor burst length**. Read-write. BIOS: 5h. Bits Description 000b 80 ns 001b 10.24 us 010b 20.48 us 011b 51.20 us 100b 76.80 us 101b 102.4 us 110b 128 us 111b 153.6 us
7:4	**Threshold: interrupt rate monitor PC6 threshold**. Read-write. BIOS: 4h. Specifies the minimum threshold the interrupt counter must reach before cache-flush is denied.
3:0	**MaxDepth: interrupt rate monitor maximum counter depth**. Read-write. BIOS: 5h. Specifies the saturation point of the interrupt counter.

D18F4x1C0 Node Cac Register 1

Bits	Description
31:12	Reserved.
11:0	**NodeCacLatest**. Read-only; Updated-by-hardware. Reset: 0. Specifies the sum of all instantaneous power credits on each compute unit. NodeCacLatest is reset to 0 when D18F4x15C[ApmMasterEn] == 0.

D18F4x250 TDP Limit 8

Bits	Description
31	Reserved.
30:28	**TdpLimitPstate**. Read-write. Reset: 0. Specifies the highest performance P-state that has a power consumption less than or equal to the TDP limit. This field is programmed by BIOS and uses software P-state numbering.
27:16	Reserved.
15:0	**NodeTdpLimit**. Read-write; Same-for-all. Reset: Product-specific. Specifies the maximum allowed sum of TDPs from all cores on a node. If the consumed power exceeds the NodeTdpLimit, a P-state limit is applied to all cores on the processor to reduce the power consumption so that it remains within the TDP limit. If D18F4x15C[BoostLock] == 1, NodeTdpLimit can only be written with values that are less than or equal to the reset value. Attempts to write an invalid value are ignored. See 2.5.8.2 [Thermal Limiting].

D18F4x2[D4:D0] Power Management Override Register 1,0

Read-write. Reset: 0000_0000h.

Table 194: Register Mapping for D18F4x2[D4:D0]

Register	Function
D18F4x2D0	Power Management Override Register 0
D18F4x2D4	Power Management Override Register 1

Bits	Description
31:23	Reserved.
22	**PwrMgtOvrdRstOnAbrt**. 1=Clear grant on an abort event.
21	**PwrMgtOvrdSticky**. 1=Does not clear the grant after a success pick or an abort event if enabled.
20:16	**PwrMgtOvrdSel**. Specifies the encoded select for override. Value Description 07h-00h Reserved 08h NbPstateValHi 1Fh-09h Reserved.
15:0	**PwrMgtOvrdGnt**. Specifies the grant vector for the selected override.

3.15 Device 18h Function 5 Configuration Registers

See 3.1 [Register Descriptions and Mnemonics]. See 2.7 [Configuration Space].

D18F5x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: device ID**. Read-only. Value: 15B5h.
15:0	**VendorID: vendor ID**. Read-only. Value: 1022h.

D18F5x04 Status/Command

Bits	Description
31:16	**Status**. Read-only. Value: 0000h.
15:0	**Command**. Read-only. Value: 0000h.

D18F5x08 Class Code/Revision ID

Bits	Description
31:8	**ClassCode**. Read-only. Value: 06_0000h. Provides the host bridge class code as defined in the PCI specification.
7:0	**RevID: revision ID**. Read-only. Value: 00h.

D18F5x0C Header Type

Bits	Description
31:0	**HeaderTypeReg**. Read-only. Reset: 0080_0000h. These bits are fixed at their default values. The header type field indicates that there are not multiple functions present in this device.

D18F5x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapPtr: capabilities pointer**. Read-only. Value: 00h.

D18F5x[70,60,50,40] Northbridge Performance Event Select Low

Bits	Description
31:0	MSRC001_024[6,4,2,0][31:0] is an alias of D18F5x[70,60,50,40].

D18F5x[74,64,54,44] Northbridge Performance Event Select High

Bits	Description
31:0	MSRC001_024[6,4,2,0][63:32] is an alias of D18F5x[74,64,54,44].

D18F5x[78,68,58,48] Northbridge Performance Event Counter Low

See 2.6.1.2 [NB Performance Monitor Counters] for proper read sequence.

Bits	Description
31:0	MSRC001_024[7,5,3,1][31:0] is an alias of D18F5x[78,68,58,48].

D18F5x[7C,6C,5C,4C] Northbridge Performance Event Counter High

See 2.6.1.2 [NB Performance Monitor Counters] for proper read sequence.

Bits	Description
31:0	MSRC001_024[7,5,3,1][63:32] is an alias of D18F5x[7C,6C,5C,4C].

D18F5x80 Compute Unit Status 1

See 2.4.4 [Processor Cores and Downcoring].
Software associates core ID to the cores of the compute units according to the following table. All combinations not listed are reserved.

Table 195: D18F5x80[Enabled, DualCore] Definition

Enabled	DualCore	Definition
1h	1h	1 compute unit is enabled; both cores of the compute unit are enabled.

Bits	Description
31:24	Reserved.
23:16	**DualCore: both cores of a compute unit are enabled**. Read-only. Reset: Product-specific. 1=Both cores of a compute unit are enabled. See Table 195 [D18F5x80[Enabled, DualCore] Definition]. Bit Description [0] Compute unit 0 [7:1] Reserved
15:8	Reserved.
7:0	**Enabled: at least one core of a compute unit is enabled**. Read-only. Reset: Product-specific. 1=At least one core is enabled in a compute unit. See Table 195 [D18F5x80[Enabled, DualCore] Definition]. Bit Description [0] Compute unit 0 [7:1] Reserved

D18F5x84 Northbridge Capabilities 2

Unless otherwise specified, 1=The feature is supported by the processor; 0=The feature is not supported.

Bits	Description
31:29	Reserved.
28:24	**DdrMaxRateEnf: enforced maximum DDR rate**. Read-only. Value: Product-specific. See: DdrMaxRate. Specifies the maximum DRAM data rate that the processor is designed to support. Writes to D18F2x94_dct[0][MemClkFreq] that specify a frequency greater than specified by DdrMaxRateEnf will result in the D18F2x94_dct[0][MemClkFreq] being set to DdrMaxRateEnf.
23:21	Reserved.
20:16	**DdrMaxRate: maximum DDR rate**. Read-only. Value: Product-specific. Specifies the maximum DRAM data rate that the processor is designed to support. DdrMaxRate is defined by Table 118 [Valid Values for Memory Clock Frequency Value Definition]; except that 00h is defined as no limit. See D18F2x94_dct[0][MemClkFreq], and DdrMaxRateEnf.
15:12	**DctEn[3:0]: DCT[3:0] enabled**. Read-only. Value: Product-specific. Specifies which DCT controllers are enabled. 1=Enabled. 0=Disabled. Bit — Description [0] — DCT 0 [3:1] — Reserved.
11:8	Reserved.
7:0	**CmpCap: CMP capable**. Read-only. Value: Product-specific. Number of cores on the node is CmpCap+1. CmpCap does not reflect cores disabled by D18F3x190[DisCore].

D18F5x88 NB Configuration 4 (NB_CFG4)

Bits	Description
31:25	Reserved.
24	**DisHbNpReqBusLock**. Read-write. Reset: 0. BIOS: 1. 0=While bus locks are in progress, all non-posted commands from IO, including atomics, are blocked until the core has completed the locked transaction and releases the bus. 1=All non-posted commands except atomics do not honor bus locks and are allowed to proceed. This bit may be set to achieve better DMA performance in the presence of bus locks.
23:19	Reserved.
18	**EnCstateBoostBlockCC6Exit**. Read-write. Reset: 0. BIOS: 1. 1=Cores cannot exit CC6 until VDD is less than or equal to the voltage of the P-state indexed by D18F4x16C[CstateBoost].
17:15	Reserved.
14	**DisHldRegRdRspChk**. Read-write. Reset: 0. BIOS: 1. 1=Disable primary holding register CPU or IO read response checks.
13:0	Reserved.

D18F5x8C NB Configuration 5 (NB_CFG5)

Bits	Description
31:27	Reserved.
26	**DisSrqPickNcReqThrtl**. Read-write. Reset: 1. BIOS: 1. 1=Disable throttling SRQ picker for requests sourced from the links.
25:16	Reserved.
15	**EnSrqAllocGt31**. Read-write. Reset: 0. BIOS: 1. 1=Enables allocation of SRA entries to above the lower 32 entries.
14:0	Reserved.

D18F5xA0 Northbridge Configuration 8(NB_CFG8)

Reset: 0000_0001h

Bits	Description
31	**DisCasBasedOpbdMgmt**. Read-write.
30:0	Reserved.

D18F5xE0 Processor TDP Running Average

Bits	Description
31:4	Reserved.
3:0	**RunAvgRange: running average range**. Read-write; Same-for-all. Reset: 0. BIOS: 2h. Specifies the interval over which the processor averages power consumption estimates from the cores for boosting. Time interval = 2^(RunAvgRange + 1) * FreeRunSampleTimer rate. A value of 0 disables the TDP running average accumulator capture function. See 2.5.8 [Application Power Management (APM)].

D18F5xE8 TDP Limit 3

Bits	Description
31:16	**ApmTdpLimit**. Read-only; Updated-by-hardware. Value: D18F4x250[NodeTdpLimit]. If the consumed node power exceeds the ApmTdpLimit on an single node processor or the ApmTdpLimit/2 on a multi-node processor, a P-state limit is applied to all cores on all nodes to reduce the power consumption to remain within the TDP limit. See 2.5.8.2 [Thermal Limiting].
15:10	**Tdp2Watt[5:0]**. Read-only. Value: Product-specific. See Tdp2Watt[15:6].
9:0	**Tdp2Watt[15:6]**. Read-only. Value: Product-specific. Specifies in watts/TDP units the conversion factor for converting TDP units to watts. Tdp2Watt[15:0] is a fixed point integer with 16 bits to the right of the decimal point and 0 bits to the left of the decimal point (e.g., Tdp2Watt[15]=0.5 W; Tdp2Watt[6]=0.976 mW; Tdp2Watt[0]=15.2 uW).

D18F5xEC Load Step Throttle Control

Bits	Description
31:19	Reserved.
18:7	**LSCacThreshold: load step Cac threshold**. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: Product-specific. Specifies the power consumption threshold required for load step throttling. D18F4x1C0[NodeCacLatest] must be less than LSCacThreshold prior to load step throttling.
6:4	**LSPstate: load step P-state**. Reset: Product-specific. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Specifies the P-state threshold required for load step throttling. This field uses hardware P-state numbering. See 2.5.2.1.1.2 [Hardware P-state Numbering].
3:1	**LSCpNum: load step compute unit number**. Reset: Product-specific. IF D18F4x15C[BoostLock] THEN Read-only. ELSE Read-write. ENDIF. Specifies the compute unit threshold required for load step throttling. The number of compute units in C0 must be greater than LSCpNum prior to load step throttling. See LSPstate.
0	**LSThrottleEn: load step throttle enable**. Reset: Product-specific. Read-write. 1=Enable the load step throttle controllers when the requirements in LSCpNum, LSPstate, and LSCacThreshold are met.

D18F5x128 Clock Power/Timing Control 3

Bits	Description
31	Reserved.
30	**NbFidChgCpuOpEn**. Read-write. Cold reset: 0. BIOS: 1.
29:28	Reserved.
27	**SprSaveRestoreEn**. Read-write. Cold Reset:0. BIOS: 1. Enables SPR save/restore for non-retention NB power gating.
26:23	Reserved.
22	**NbPllPwrDwnRegEn: NB PLL power down**. Read-write. Cold reset: Product-specific. 1=The NB PLL is powered down when the NB is power gated and DRAM is placed into self-refresh (see 2.5.3.2 [NB C-states]). 0=The NB PLL is not powered down during NB C-states.
21	**PC6Vid[7]**. Read-write. Cold reset: Product-specific. See PC6Vid[6:0].
20:16	Reserved.
15	**CC6PwrDwnRegEn: CC6 power down regulator enable**. Read-write. Cold reset: Product-specific. 1=Power down the VDDA regulator on CC6 entry. See PllRegTime.
14	**PC6PwrDwnRegEn: PC6 power down regulator enable**. Read-write. Cold reset: Product-specific. 1=Power down the VDDA regulator on PC6 entry. See PllRegTime.
13:12	**PwrGateTmr: power gate timer**. Read-write. Cold reset: 01b. BIOS: 01b. Specifies the minimum delay time required from the power gating or ungating of one Compute Unit to the power gating or ungating of the same Compute Unit or another Compute Unit. Bits Description Bits Description 00b 500 ns 10b Reserved. 01b 1 us 11b Reserved.

11:10	**PllVddOutUpTime**. Read-write. Cold reset: 0. The VDD regulator may be powered down when the processor transitions to PC6. If the regulator is powered down, this field specifies the time required to initialize the core PLL logic once the regulator is powered back up. Bits Description Bits Description 00b 100 ns 10b 400 ns 01b 200 ns 11b 800 ns
9	**FastSlamTimeDown**. Read-write. Cold reset: 0. BIOS: 1. Specifies the time the processor waits for downward voltage transitions to complete. This field only effects transitions from D18F4x16C[CstateBoost] or lower performance P-states. 0=D18F3xD8[VSRampSlamTime]. 1=10 us.
8:7	**PllRegTime: Pll regulator time**. Read-write. Cold reset: 10b. The VDDAregulator may be powered down when the processor transitions to PC6 or CC6. See PC6PwrDwnRegEn and CC6PwrDwnRegEn. If CC6PwrDwnRegEn == 1, the VDDA regulator is powered down during CC6. If PC6PwrDwnRegEn == 1, the VDDA regulator is powered down during PC6. If the VDDA regulator is powered down during CC6 and the core transitions from CC6 to PC6, the regulator remains powered down during PC6 regardless of the PC6PwrDwnRegEn setting. This field specifies the time required for the VDDA regulator to power back up and initialize the core PLL logic that is powered by the VDDA regulator. Bits Description Bits Description 00b Reserved. 10b 1.5 us 01b Reserved. 11b 2.0 us
6:0	**PC6Vid[6:0]: package C6 vid**. Read-write. Cold reset: Product-specific. PC6Vid[7:0] = {PC6Vid[7], PC6Vid[6:0]}. PC6Vid[7:0] specifies the VID driven in the PC6 state. See 2.5.2.2.3.4 [Package C6 (PC6) State].

D18F5x16[C:0] Northbridge P-state [3:0]

Each of these registers specify the frequency and voltage associated with each of the NB P-states.

Table 196: Register Mapping for D18F5x16[C:0]

Register	Function
D18F5x160	NB P-state 0
D18F5x164	NB P-state 1
D18F5x168	NB P-state 2
D18F5x16C	NB P-state 3

The NbVid field is allowed to be different between processors in a multi-processor system. All other fields are required to be programmed to the same value for all processors in the coherent fabric. See 2.5.3.1 [NB P-states] for more information about these registers.

Table 197: NB P-state Definitions

Term	Definition
NBCOF	NB current operating frequency in MHz. NBCOF = 100 * (D18F5x16[C:0][NbFid] + 4h) / (2^D18F5x16[C:0][NbDid]).
NBCOF[0]	NB current operating frequency in MHz for NB P-state 0. NBCOF[0] = (100 * (D18F5x160[NbFid] + 4h) / (2^D18F5x160[NbDid])).

Table 197: NB P-state Definitions (Continued)

Term	Definition
NBCOF[1]	NB current operating frequency in MHz for NB P-state 1. NBCOF[1] = (100 * (D18F5x164[NbFid] + 4h) / (2^D18F5x164[NbDid])).
NBCOF[2]	NB current operating frequency in MHz for NB P-state 2. NBCOF[2] = (100 * (D18F5x168[NbFid] + 4h) / (2^D18F5x168[NbDid])).
NBCOF[3]	NB current operating frequency in MHz for NB P-state 3. NBCOF[3] = (100 * (D18F5x16C[NbFid] + 4h) / (2^D18F5x16C[NbDid])).

Bits	Description
31:24	**NbIddValue: Northbridge current value**. Read-write. Reset: Product-specific. See NbIddDiv.
23:22	**NbIddDiv: Northbridge current divisor**. Read-write. Reset: Product-specific. After reset, NbIddDiv and NbIddValue combine to specify the expected maximum current drawn on the VDDNB power plane at a given VDDNB voltage. These values are intended to be used by 2.5.1.3.1.1 [BIOS Requirements for PSI0_L]. These values are not intended to convey final product power levels and may not match the power levels specified in the Power and Thermal Datasheet. These fields may be subsequently altered by software; they do not affect the hardware behavior. Bits — Description 00b — IddValue / 1 A, Range: 0 to 255 A. 01b — IddValue / 10 A, Range: 0 to 25.5 A. 10b — IddValue / 100 A, Range: 0 to 2.55 A. 11b — Reserved.
21	**NbVid[7]**. Read-write. Reset: Product-specific. See NbVid[6:0].
20:19	Reserved.
18	**MemPstate: Memory P-state**. Read-write. Reset: Product-specific. 1=The Northbridge P-state specified by this register maps to memory P-state 1. 0=The Northbridge P-state specified by this register maps to memory P-state 0. Memory P-states may be globally disabled by programming D18F5x170[MemPstateDis]. See 2.5.6.1 [Memory P-states].
17	Reserved.
16:10	**NbVid[6:0]: Northbridge VID**. Read-write. Reset: Product-specific. NbVid[7:0] = {NbVid[7], NbVid[6:0]}. NbVid[7:0] specifies the Northbridge voltage.
9:8	Reserved.
7	**NbDid: Northbridge divisor ID**. Read-write. Reset: Product-specific. Specifies the Northbridge frequency divisor. See NbFid[5:0].
6:1	**NbFid[5:0]: Northbridge frequency ID**. Read-write. Reset: Product-specific. Specifies the Northbridge frequency multiplier. The NB COF is a function of NbFid and NbDid, and defined by NBCOF. NbFid and NbDid are not changed on a write if the value written results in a frequency greater than D18F3xCC[MaxNbCof].
0	**NbPstateEn: Northbridge P-state enable**. Read-write. Reset: Product-specific. 1=The Northbridge P-state specified by this register is valid. 0=The Northbridge P-state specified by this register is not valid. This bit must be set to 1 in order for the Northbridge P-state specified by this register to be programmed in D18F5x170[NbPstateHi, NbPstateLo].

D18F5x170 Northbridge P-state Control

See also 2.5.3.1 [NB P-states].

Bits	Description
31	**MemPstateDis: memory P-state disable**. IF (D18F3xE8[MemPstateCap] && D18F2x1B4[SmuCfgLock] == 0) THEN Read-write; Updated-by-hardware; Updated-by-SMU. ELSE Read-only; Updated-by-hardware; Updated-by-SMU. ENDIF. Reset: Product-specific. 1=Memory P-state transitions are disabled. The current P-state is not changed by programming this bit. The memory P-state is forced to M0 on the next NB P-state transition. On processors where memory P-states are enabled, programming this bit may result in a violation of bandwidth requirements. Software must ensure that NB P-states which violate those requirements are forced disabled. 0=Memory P-state transitions are enabled if D18F2x90_dct[0][DisDllShutdownSR] == 0.
30	**NbPstateFidVidSbcEn**. IF (D18F5x174[NbPstateDis] \|\| D18F2x1B4[SmuCfgLock]) THEN Read-only. ELSE Read; Write-1-only. ENDIF. Reset: 0. BIOS: 1. NB P-state transitions are blocked until this field is set to a 1.
29:27	**NbPstateHiRes: NB P-state high residency timer**. If D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: 0. Specifies the minimum time the processor must spend in the high NB P-state before transitions to the low NB P-state are allowed. See 2.5.3.1 [NB P-states]. Bits / Description / Bits / Description 000b 0 us 100b 1 ms 001b 10 us 101b 5 ms 010b 100 us 110b 10 ms 011b 500 us 111b 50 ms
26:24	**NbPstateLoRes: NB P-state low residency timer**. If D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: 0. Specifies the minimum time the processor must spend in the low NB P-state before transitions to the high NB P-state are allowed. See 2.5.3.1 [NB P-states]. See: NbPstateHiRes.
23	**NbPstateGnbSlowDis**. IF D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: 0. Specifies whether NBP-state transitions take the GnbSlow signal into account. 0=Take GnbSlow into account. 1=Ignore GnbSlow. See 2.5.3.1 [NB P-states].
22:15	Reserved.
14	**SwNbPstateLoDis: software NB P-state low disable**. IF (D18F5x174[NbPstateDis] \| D18F2x1B4[SmuCfgLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 1=Transition to NbPstateHi and disable transitions to NbPstateLo.
13	**NbPstateDisOnP0: NB P-state disable on P0**. IF (D18F5x174[NbPstateDis] \| D18F2x1B4[SmuCfgLock]) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 1=Transition to NbPstateHi and disable transitions to NbPstateLo if any compute unit is in P0 or a boosted P-state. This field uses software P-state numbering. See 2.5.2.1.1.1 [Software P-state Numbering].
12:9	**NbPstateThreshold: NB P-state threshold**. IF D18F2x1B4[SmuCfgLock] THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Reset: Product-specific. BIOS: COUNT(D18F5x80[Enabled]). Specifies the minimum number of compute units that must be in a P-state with MSRC001_00[6B:64][NbPstate] == 1 before transitions to lower performance NB P-states are allowed. See NbPstateLo and NbPstateHi.
8	Reserved.

7:6	**NbPstateHi: NB P-state high**. IF (D18F2x1B4[SmuCfgLock]) THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Cold reset: Product-specific. If NB P-states are enabled, this field specifies the NB P-state that is used when the number of compute units in a P-state with MSRC001_00[6B:64][NbPstate] == 1 is less than NbPstateThreshold. This field must be programmed to the same value for all processors in the coherent fabric. This field is not changed on a write if the value written is greater than the NbPstateMaxVal value written or greater than the current NbPstateLo value. See also NbPstateDisOnP0, SwNbPstateLoDis, NbPstateLo, D18F5x174[NbPstateDis], and D18F5x16[C:0][NbPstateEn].
5	Reserved.
4:3	**NbPstateLo: NB P-state low**. IF (D18F2x1B4[SmuCfgLock]) THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Cold reset: Product-specific. If NB P-states are enabled, this field specifies the NB P-state that is used when the number of compute units in a P-state with MSRC001_00[6B:64][NbPstate] == 1 is greater than or equal to NbPstateThreshold. NbPstateLo must be greater than or equal to NbPstateHi. This field must be programmed to the same value for all processors in the coherent fabric. This field is not changed on a write if the value written is greater than the NbPstateMaxVal value written or less than the current NbPstateHi value. See also NbPstateDisOnP0, SwNbPstateLoDis, D18F5x174[NbPstateDis], and D18F5x16[C:0][NbPstateEn].
2	Reserved.
1:0	**NbPstateMaxVal: NB P-state maximum value**. IF (D18F2x1B4[SmuCfgLock]) THEN Read-only; Updated-by-hardware. ELSE Read-write. ENDIF. Cold reset: specified by the reset state of D18F5x16[C:0][NbPstateEn]; the cold reset value is the highest NB P-state number corresponding to the register in which NbPstateEn is set (e.g., if D18F5x160 and D18F5x164 have this bit set and the others do not, then NbPstateMaxVal=1; if NbPstateEn is only set in D18F5x160, then NbPstateMaxVal=0). This specifies the highest NB P-state value (lowest performance state) supported by the hardware.

D18F5x174 Northbridge P-state Status

Bits	Description
31	**NbPstateReqBusy**. Read-only; Updated-by-hardware. Reset: 0. 1=NB P-state request is pending. 0=No NB P-state requests are outstanding.
30:25	Reserved.
24	**CurMemPstate: current memory P-state**. Read-only; Updated-by-hardware. Reset: 0. Specifies the current memory P-state. 1=Memory P-state 1. 0=Memory P-state 0. See 2.5.6.1 [Memory P-states].
23	**CurNbVid[7]: current northbridge voltage ID[7]**. MSRC001_0071[CurNbVid[7]] is an alias of D18F5x174[CurNbVid[7]]. VDDNB voltage.
22	**CurNbPstateLo.** Read-only; Updated-by-hardware. Reset: 0. 1=Current NB Pstate maps to D18F5x170[NbPstateLo]. 0=Current NB Pstate maps to D18F5x170[NbPstateHi].
21	Reserved.

20:19	**CurNbPstate: current northbridge P-state**. Read-only; Updated-by-hardware. Reset: 0. Provides the NB P-state that corresponds to the current frequency component of the NB. The value of this field is updated when the COF transitions to a new value associated with an NB P-state. Bits Description 00b NB P0 01b NB P1 10b NB P2 11b NB P3
18:12	**CurNbVid[6:0]: current northbridge voltage ID**. MSRC001_0071[CurNbVid[6:0]] is an alias of D18F5x174[CurNbVid[6:0]]. VDDNB voltage.
11:10	Reserved.
9	**CurNbDid: current northbridge divisor ID**. Read-only; Updated-by-hardware. Reset: 0.
8:3	**CurNbFid[5:0]: current northbridge frequency ID**. Read-only; Updated-by-hardware. Reset: 0.
2:1	**StartupNbPstate: startup northbridge P-state number**. Read-only. Cold reset: Product-specific. Specifies the cold reset VID, FID and DID for the Northbridge based on the NB P-state number selected.
0	**NbPstateDis: northbridge P-state disable**. Read-only. Value: Product-specific. MSRC001_0071[NbPstateDis] is an alias of D18F5x174[NbPstateDis].

D18F5x178 Northbridge FCH Configuration

Bits	Description
31:20	Reserved.
19	**SwGfxDis**. Read-write. Reset: 1. BIOS: IF (GpuEnabled) THEN 0. ELSE 1. ENDIF. 1=Hardware handshakes for NB P-state transitions and DRAM self-refresh entry are ignored. SwGfxDis is disabled whenever SMU is actively controlling NB P-state plicy. See 2.5.3.1.1 [NB P-state Transitions]. See 2.5.6.2 [DRAM Self-Refresh].
18	**CstateFchHsDis: C-state fch handshake disable**. Read-write. Reset: 0. BIOS: 1. 1=Ignore the FCH handshake response for PC6 transitions. 0=Use the FCH handshake response for PC6 entry. See 2.5.2.2.4.1 [FCH Messaging].
17	**Dis2ndGnbAllowPsWait**. Read-write. Reset: 0. BIOS: 1. 1=Do not do a second check of AllowNbTrans after quiescing the cores when transitioning NB P-states. See 2.5.3.1.1 [NB P-state Transitions].
16:12	Reserved.
11	**AllowSelfRefrS3Dis: allow self-refresh S3 disable**. Read-write. Reset: 0. BIOS: 1. 1=The NB does not wait for handshake before placing DRAM into self-refresh (see 2.5.6.2 [DRAM Self-Refresh]) on S3 entry (see 2.5.7.1.1 [ACPI Suspend to RAM State (S3)]). 0=The NB waits for handshake before placing DRAM into self-refresh on S3 entry.
10	**InbWakeS3Dis: InbWake S3 disable**. Read-write. Reset: 0. BIOS: 1. 1=The NB does not wait for handshake before placing DRAM into self-refresh (see 2.5.6.2 [DRAM Self-Refresh]) on S3 entry (see 2.5.7.1.1 [ACPI Suspend to RAM State (S3)]). 0=The NB waits for handshake before placing DRAM into self-refresh on S3 entry.
9:4	Reserved.

3	**CstateThreeWayHsEn: C-state three way handshake enable**. Read-write. Reset: 0. 1=Enable the three way handshake with the FCH when entering a C-state. 0=Only a two way handshake with FCH is used. There is no message about the resulting package state sent to FCH. See 2.5.2.2.4.1 [FCH Messaging].
2	**CstateFchDis: C-state fch disable**. Read-write. Reset: 0. 1=All HALT or C-state requests are forwarded to the FCH. 0=HALT and C-state requests are forwarded to the FCH when each core has made a request. See 2.5.2.2.4.1 [FCH Messaging].
1:0	Reserved.

D18F5x1[FC:C0] BIOS Scratch

Provides the BIOS, Microcode, or Software with general purpose scratch registers. Hardware will not use or modify these registers and will clear these registers only at Cold reset.

Table 198: Register Mapping for D18F5x1[FC:C0]

Register	Function
D18F5x1[D4:C0]	Reserved.
D18F5x1[FC:D8]	BIOS Scratch

Bits	Description
31:0	**Scratch**. Read-write. Reset: 0.

D18F5x21[8:4] ONION3 Link Controller Buffer Count

Table 199: Register Mapping for D18F5x21[8:4]

Register	Function
D18F5x214	Link 0
D18F5x218	Link 1

Bits	Description
31:26	Reserved.
25:24	**On3DatPrivateLmt**. Read-write. Reset: 0. BIOS: 2h. Specifies the number of inbound ONION3 to MCT private data token limit for the ONION3 link.
23:18	Reserved.
17:16	**On3ReqPrivateLmt**. Read-write. Reset: 0. BIOS: 3h. Specifies the number of inbound ONION3 to MCT private request token limit for the ONION3 link.
15:13	Reserved.
12:8	**On3FreeDatLmt**. Read-write. Reset: 04h. Specifies the number of inbound ONION3 to MCT freelist data token limit for the ONION3 link.
7:5	Reserved.
4:0	**On3FreeCmdLmt**. Read-write. Reset: 04h. BIOS: 05h. Specifies the number of inbound ONION3 to MCT freelist request token limit for the ONION3 link.

D18F5x224 Extended MCT Buffer Count

Bits	Description
31	Reserved.
30:28	**On3UrgCmdMax**. Read-write. Reset: 0. BIOS: 7h. Selects the number of MCQ tokens allocated to an urgent channel for every token allocated to each non-urgent channel. This field has to be set to a value greater than 0 to enable the feature.
27	Reserved.
26:24	**On3UrgDatMax**. Read-write. Reset: 0. BIOS: 4. Selects the number of MCD tokens allocated to an urgent channel for every token allocated to each non-urgent channel. This field has to be set to a value greater than 0 to enable the feature.
23:18	Reserved.
17:16	**XbartoMctWrbkDatCmdLmt**. Read-write. Reset: 0. Specifies the number of XBAR to MCT hard-allocated (private) write back tokens. Bits Description 00b Default 01b Reserved. 10b Reserved. 11b Reserved.
15:14	Reserved.
13:12	**XbartoMctDatPrivateLmt**. Read-write. Reset: 0. BIOS: 2. Specifies the number of XBAR to MCT hard-allocated (private) non-isoc data tokens.
11:10	Reserved.
9:8	**XbartoMctReqPrivateLmt**. Read-write. Reset: 0. BIOS: 2h. Specifies the number of XBAR to MCT hard-allocated (private) non-isoc command tokens. Bits Description 00b Reserved. 01b Reserved. 10b 2 Tokens. 11b 3 Tokens
7	Reserved.
6:4	**XbartoMctFreeDatLmt**. Read-write. Reset: 4h. Specifies the number of XBAR to MCT freelist data tokens.
3	Reserved.
2:0	**XbartoMctFreeCmdLmt**. Read-write. Reset: 4h. Specifies the number of XBAR to MCT freelist command tokens.

D18F5x228 MCT to ONION3 Buffer Count

Bits	Description
31:25	Reserved.
24:21	**FreeOpbdDcqStallThresh**. Read-write. Reset: 4h. BIOS: 3h. Set the DRAM command queue stall threshold.

20:16	**McttoOn3RspData**. Read-write. Reset: 12h. BIOS: 12h. Specifies the number of Response data tokens in the MCT available for sending responses to the ONION3 controller. Bits: Description 07h-00h: Reserved. 12h-08h: Number of Response data tokens available. 1Fh-13h: Reserved.
15:11	Reserved.
10:8	**McttoOn3Data**. Read-write. Reset: 5h. Specifies the number of MCT to ONION3 controller out-bound data tokens.
7:5	Reserved.
4:0	**McttoOn3Cmd**. Read-write. Reset: 10h. Specifies the number of MCT to ONION3 controller out-bound command tokens.

D18F5x22C ONION3 Configuration

Bits	Description
31:25	**McqOn3PrbRspDly**. Read-write. Reset: 0. Holds the number of NCLK cycles predicted to cover the time it takes the GNB to execute an ONION3 probe. Bits: Description 00h: Disables the ONION3 Probe Response Predictor. 7Fh-01h: Reserved.
24:20	**On3CfgMctWrLmt**. Read-write. Reset: 1Fh. Specifies the number of ONION3 DRAM writes in the memory controller queue before they are burst into the DCTs. Disabling Write Burst is recommended for GPU intensive applications. Due to the time to drain the MCQ, burst sizes should be no larger than 12 entries, or a setting of 14h. Bits: Description 00h: 32 1Dh-01h: [32-On3CfgMctWrLmt] 1Eh: 2 1Fh: Write bursting disabled
19:16	**On3PrbLnkMap**. Read-write. Reset: Ch. BIOS: Ah. Selects the link to be probed depending on the decode of the XBAR to MCT Address Map. Bit: Description [0]: Selects the link to be probed if XbarToMctAddr[9:8] = 00b. 1=Link1, 0=Link0. [1]: Selects the link to be probed if XbarToMctAddr[9:8] = 01b. 1=Link1, 0=Link0. [2]: Selects the link to be probed if XbarToMctAddr[9:8] = 10b. 1=Link1, 0=Link0. [3]: Selects the link to be probed if XbarToMctAddr[9:8] = 11b. 1=Link1, 0=Link0.
15:11	Reserved.
10	**DisDatFWOn3**. Read-write. Reset: 1. 0=Reserved.
9:7	**MsbClrStallTh**. Read-write. Reset: 1.
6:4	**MsbSetStallTh**. Read-write. Reset: 3.
3:1	**McqOn3RtUrgPriByPassMax**. Read-write. Reset: 4h. Specifies the number of ONION3 real-time urgent ops that can bypass all other ops in the MCQ, before the picker chooses a non ONION3 real-time urgent op. This field must be programmed with a non-zero value.
0	Reserved.

GMMx2024 MC_VM_FB_LOCATION

Read-write. Reset: 0000_0000h. This register specifies the location of the frame buffer in the internal address space. The internal address space has 40 address bits. The minimum frame buffer size is 16 MB, and the start location is required to be on a 16 MB boundary. Therefore BASE(23:0) must be 0x000000 and TOP(23:0) must be 0xFFFFFF. Only the 16 MSBs of each are loaded in the register. If inside the TOP/BASE aperture, the address is corrected by subtracting BASE. The register fields define bits (39:24) for both BASE and TOP.

Bits	Description
31:16	**FbTop**.
15:0	**FbBase**.

GMMx2068 MC_VM_FB_OFFSET

Read-write. Reset: 0000_0000h. This register defines the physical location of FB memory in UMA physical memory for the APU. The system physical and internal address spaces have 40 address bits. The minimum FB size is 16 MB, and the start location is required to be on a 16 MB boundary. Therefore, BASE(23:0) is treated as 0x000000. Only the 16 MSBs of each are loaded in the register. If inside the FBTOP/FBBOT aperture, the address is adjusted by subtracting FBBOT and adding FB_OFFSET. The register field defines bits (39:22) for FB_OFFSET. Only bits 39:24 are applicable as the minimum frame buffer size is 16MB. The lowest 2 bits of this register are not used.

Bits	Description
31:18	Reserved.
17:0	**FbOffset**.

GMMx224[C:4] MC_VM_MB_L1_DEBUG

Table 200: Register Mapping for GMMx224[C:4]

Register	Function
GMMx2244	TLB0
GMMx2248	TLB1
GMMx224C	TLB2

Bits	Description
31:19	Reserved.
18:15	**L1_TLB_DEBUG[3:0]**. Read-write. Reset: Ah. Bit — Definition [0] — Enable HDP aperture fault in MCB. [1] — Reserved. [2] — Reserved. [3] — Reserved. See 2.14.2 [Frame Buffer (FB)].
14:0	Reserved.

3.16 Northbridge IOAPIC Registers

The Northbridge IOAPIC is accessed through the Northbridge IOAPIC base address specified by D0F0xFC_x01 [IOAPIC Base Address Lower] and D0F0xFC_x02 [IOAPIC Base Address Upper].

NBIOAPICx00 IO Register Select

Bits	Description
31:8	Reserved.
7:0	**IndirectAddressOffset**. Read-write. Reset: 0. Specifies the indexed register accessed via NBIOAPICx10 [IO Window].

NBIOAPICx10 IO Window

Bits	Description
31:0	**IoapicData**.

NBIOAPICx10_x00 IOAPIC ID

This register is not used in IOxAPIC PCI bus delivery mode.

Bits	Description
31:28	**ExtendID: extended IOAPIC device ID**. IF (D0F0xFC_x00[IoapicIdExtEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.
27:24	**ID: IOAPIC device ID**. Read-write. Reset: 0.
23:0	Reserved.

NBIOAPICx10_x01 IOAPIC Version

Bits	Description
31:24	Reserved.
23:16	**MaxRedirectionEntries**. Value: 1Fh. Indicates 32 entries [31:0].
15	**PRQ**. Value: 1. IRQ pin assertion supported.
14:8	Reserved.
7:0	**Version**. Value: 21h. PCI 2.2 compliant.

NBIOAPICx10_x02 IOAPIC Arbitration

Bits	Description
31:28	Reserved.
27:24	**ArbitrationID**. Read-only. Reset: 0.
23:0	Reserved.

NBIOAPICx10_x[4E:10:step2] Redirection Table Entry [31:0]

Bits	Description
63:56	**DestinationID**. Read-write. Reset: 0. Bits[19:12] of the address field of the interrupt message.
55:17	Reserved.
16	**Mask**. Read-write. Reset: 1. 1=Mask the interrupt injection at the input of this device. 0=Unmask.
15	**TriggerMode**. Read-write. Reset: 0. 0=Edge. 1=Level.
14	**RemoteIRR**. Read-only. Reset: 0. Used for level triggered interrupts only. It is cleared by EOI special cycle transaction or write to EOI register. 1=Interrupt message is delivered.
13	**InterruptPinPolarity**. Read-write. Reset: 0. 0=High. 1=Low.
12	**DeliveryStatus**. Read-only. Reset: 0. 0=Idle. 1=Send pending.
11	**DestinationMode**. Read-write. Reset: 0. 0=Physical. 1=Logical.
10:8	**DeliveryMode**. Read-write. Reset: 0. Bits Definition Bits Definition 000b Fixed 100b NMI 001b Lowest Priority 101b INIT 010b SMI/PMI 110b Reserved 011b Reserved 111b ExtINT
7:0	**Vector**. Read-write. Reset: 0. Interrupt vector associated with this interrupt input.

NBIOAPICx20 IRQ Pin Assertion

Bits	Description
31:8	Reserved.
7:0	**InputIrq**. Read-write. Reset: 0. IRQ number for the requested interrupt. A write to this register triggers an interrupt associated with the redirection table entry referenced by the IRQ number. Currently the redirection table has 24 entries. Writes with IRQ number greater than 17h have no effect.

NBIOAPICx40 EOI

Bits	Description
31:8	Reserved.
7:0	**Vector**. Write-only. Reset: 0. Interrupt vector. A write to this register clears the remote IRR bit in the redirection table entry found matching the interrupt vector. This provides an alternate mechanism other than PCI special cycle for EOI to reach IOxAPIC.

3.17 IOMMU Memory Mapped Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.12.1 [IOMMU Configuration Space].

IOMMUx00 Device Table Base Address Low

Bits	Description
31:12	**DevTblBase[31:12]: device table base address bits[31:12]**. Read-write. Reset: 0. DevTblBase[51:12] = {IOMMUx04[DevTblBase[51:32], DevTblBase[31:12]]}. DevTblBase[51:12] specifies the 4Kbyte-aligned base address of the first level device table.
11:9	Reserved.
8:0	**DevTblSize: device table size**. Read-write. Reset: 0. This field contains 1 less than the length of the device table, in multiples of 4K bytes. A minimum size of 0 corresponds to a 4K byte device table and a maximum size of 1FFh corresponds to a 2M byte device table.

IOMMUx04 Device Table Base Address High

Bits	Description
31:20	Reserved.
19:0	**DevTblBase[51:32]: device table base address bits[51:32]**. See: IOMMUx00[DevTblBase[31:12]].

IOMMUx08 Command Buffer Base Address Low

Bits	Description
31:12	**ComBase[31:12]: command buffer base address bits[31:12]**. Read-write. Reset: 0. ComBase[51:12] = {IOMMUx0C[ComBase[51:32], ComBase[31:12]]}. ComBase[51:12] specifies the 4Kbyte-aligned base address of the command buffer.
11:0	Reserved.

IOMMUx0C Command Buffer Base Address High

Bits	Description
31:28	Reserved.
27:24	**ComLen: command buffer length**. Read-write. Reset: 8h. Specifies the length of the command buffer in power of 2 increments. The minimum size is 256 entries (4K bytes); values less than 8h are reserved. Bits — Description 7h-0h — Reserved. Fh-8h — 2^ComLen entries (2^ComLen*16 bytes).
23:20	Reserved.
19:0	**ComBase[51:32]: command buffer base address bits[51:32]**. See: IOMMUx08[ComBase[31:12]].

IOMMUx10 Event Log Base Address Low

Bits	Description
31:12	**EventBase[31:12]: event log base address bits[31:12]**. Read-write. Reset: 0. EventBase[51:12] = {IOMMUx14[EventBase[51:32], EventBase[31:12]]}. EventBase[51:12] specifies the 4K-byte aligned base address of the event log.
11:0	Reserved.

IOMMUx14 Event Log Base Address High

Bits	Description
31:28	Reserved.
27:24	**EventLen: event log length**. Read-write. Reset: 8h. Specifies the length of the event log in power of 2 increments. The minimum size is 256 entries (4K bytes); values less than 8h are reserved. Bits — Description 7h-0h — Reserved. Fh-8h — 2^EventLen entries (2^EvemtLen*16 bytes).
23:20	Reserved.
19:0	**EventBase[51:32]: event log base address bits [51:32]**. See: IOMMUx10[EventBase[31:12]].

IOMMUx18 Control Low

Bits	Description
31:30	**PprQ.** Read-write. Reset: 0. Bits — Description 00b — Use default PPR log queue for PPR logger with autoswap disabled. 01b — Use PPR logBqueue for PPR logger with autoswap disabled. 10b — Autoswap queue between default PPR log and PPRlog. Always start with default PPR log after reset. 11b — Unused.
29	**GaIntEn.** Read-write. Reset: 0.
28	**GaLogEn.** Read-write. Reset: 0.
27:25	**GamEn.** Read-write. Reset: 0.
24	**SmiFLogEn: SMI filter log enable**. Read-write. Reset: 0. Specifies if blocked SMI interrupts are reported in the IOMMU event log. When SmiFSup == 00b, SmiFLogEn is ignored by hardware and may be implemented as a read-only value of 0. 0=SMI interrupts are not logged in the IOMMU event log (same behavior as IOMMU Revision 1). 1=SMI interrupts blocked due to a match-failure with all valid (SmiDV == 1) SMI filter registers are reported in the IOMMU event log.
23	**GaP2pWriteDis.** Read-write. Reset: 0.
22	**SmiFEn: SMI filter enable**. Read-write. Reset: 0. Specifies how SMI interrupts are controlled by the IOMMU. When SmiFSup == 00b, SmiFEn is ignored by hardware and may be implemented as a read-only value of 0. 0=SMI interrupts are always passed-through (same behavior as IOMMU Revision 1). 1=SMI interrupts are blocked unless otherwise controlled by the SMI Filter Registers and blocked SMI interrupts are reported in the event log as governed by SmiFLogEn.

21:18	**Tlpt**. Read-write. Reset: 0. Tlpt contains the 4-bit value matched to the PCIe® TLP Type field when the PCIe TLP Fmt value indicates the field carries a prefix.
17	**GaEn**. Read-write. Reset: 0. Guest APIC enable. 1=Loose. 0=Prohibited.
16	**GtEn**. Read-write. Reset: 0. 1=Guest translation may be enabled for a peripheral by programming DTE[GV]. This bit must be programmed to zero when IOMMUx30[GtSup] == 0.
15	**PprEn**. Read-write. Reset: 0. 1=Peripheral page service requests are processed. 0=Peripheral page service requests are treated as invalid device requests. This bit must be programmed to zero when IOMMUx30[PprSup] == 0.
14	**PprIntEn**. Read-write. Reset: 0. 1=An interrupt is generated when IOMMUx2020[PprInt] == 1 or IOMMUx2020[PprOverflow] == 1. The interrupt vector used is indicated in D0F2x50[IommuMsiNumPpr].This bit must be programmed to zero when IOMMUx30[PprSup] == 0.
13	**PprLogEn**. Read-write. Reset: 0. 1=Peripheral page service request events are written to the PPR log when IommuEn == 1. 0=Peripheral page service request logging is not enabled. Peripheral page service requests are discarded when PprLogEn == 0 or IOMMUx30[PprSup] == 0. When IommuEn == 1 and software sets PprLogEn, the IOMMU clears IOMMUx2020[PprOverflow] and sets IOMMUx2020[PprRun]. The IOMMU can then write new entries to the event log if there are usable entries available. Software can read IOMMUx2020[PprRun] to determine the status of the peripheral page service request log. Note the peripheral page service request and event logs are independent. IOMMUx38, IOMMUx2030, and IOMMUx2038 must be programmed prior to setting PprLogEn.
12	**CmdBufEn**. Read-write. Reset: 0. 1=Start or restart command buffer processing. When CmdBufEn == 1 and IommuEn == 1, the IOMMU starts fetching commands and sets IOMMUx2020[CmdBufRun]. 0=Halt command buffer processing. Writing a 0 to this bit causes the IOMMU to cease fetching new commands although commands previously fetched are completed. The IOMMU stops fetching commands upon reset and after errors. See IOMMUx2020[CmdBufRun]. Writing of event log entries is independently controlled by EventLogEn. IOMMUx08, IOMMUx0C, IOMMUx2000, and IOMMUx2008 must be programmed prior to setting CmdBufEn.
11	**Isoc**. Read-write. Reset: 0. This bit controls the state of the isochronous bit in the HyperTransport™ read request packet when the IOMMU issues IO page table reads and device table reads on the HyperTransport link. 1=Request packet to use isochronous channel. 0=Request packet to use standard channel. If IOMMU isoc requests are enabled, then the isoc channel must be enabled as well. See D0F0x98_x1E[HiPriEn].
10	**Coherent**. Read-write. Reset: 1. This bit controls the state of the coherent bit in the HyperTransport™ read request packet when the IOMMU issues device table reads on the HyperTransport link. 1=Device table requests are snooped by the processor. 0=Device table requests are not snooped by the processor.
9	**ResPassPw**. Read-write. Reset: 0. This bit controls the state of the ResPassPW bit in the HyperTransport read request packet when the IOMMU issues IO page table reads and device table reads on the HyperTransport link. 1=Response may pass posted requests. 0=Response may not pass posted requests.
8	**PassPw**. Read-write. Reset: 0. This bit controls the state of the PassPW bit in the HyperTransport™ read request packet when the IOMMU issues IO page table reads and device table reads on the HyperTransport link. 1=Request packet may pass posted requests. 0=Request packet may not pass posted requests.

7:5	**InvTimeout**. Read-write. Reset: 0. This field specifies the invalidation timeout for IOTLB invalidation requests. Bits: Description 000b: No timeout. 001b: 1 ms. 010b: 10 ms. 011b: 100 ms. 100b: 1 sec. 101b: 10 sec. 111b-110b: Reserved
4	**ComWaitIntEn**. Read-write. Reset: 0. 1=An interrupt is generated when IOMMUx2020[ComWaitInt] == 1.
3	**EventIntEn**. Read-write. Reset: 0. 1=An interrupt is generated when IOMMUx2020[EventLogInt] == 1 or IOMMUx2020[EventOverflow] == 1.
2	**EventLogEn**. Read-write. Reset: 0. 1=All events detected are written to the event log when IommuEn == 1. 0=Event logging is not enabled. Events are discarded when the event log is not enabled. When IommuEn == 1 and software sets EventLogEn, the IOMMU clears IOMMUx2020[EventOverflow] and sets IOMMUx2020[EventLogRun]. IOMMUx10, IOMMUx14, IOMMUx2010, and IOMMUx2018 must be programmed prior to setting EventLogEn.
1	**HtTunEn**. Read-write. Reset: 0. 1= Upstream traffic received by the HyperTransport™ tunnel is translated by the IOMMU. 0=Upstream traffic received by the HyperTransport tunnel is not translated by the IOMMU. The IOMMU ignores the state of this bit while IommuEn == 0. See D0F2x40[IommuHtTunnelSup].
0	**IommuEn**. Read-write. Reset: 0. 1=IOMMU enabled. All upstream transactions are translated by the IOMMU. IOMMUx00 [Device Table Base Address Low] and IOMMUx04 [Device Table Base Address High] must be configured by software before setting this bit. 0=IOMMU is disabled and no upstream transactions are translated or remapped by the IOMMU. When disabled, the IOMMU does not read any commands or create any event log entries.

IOMMUx1C Control High

Bits	Description
31:11	Reserved.
10	**PprAutoRespAOn**. Read-write. Reset: 0. Only effective when IOMMUx1C[PprAutoRespEn] == 1. Generate PPR auto response all the time regardless of the settings in IOMMUx2088[PprOverflowEarlyThres] and IOMMUx2090[PprBOverflowEarlyThres].
9	**BlockStopMarkEn**. Read-write. Reset: 1. Blocking stop mark messages.
8	**MarcEn**. Read-write. Reset: 0. Enable the memory access routing and control feature.
7	**PprAutoRespEn**. Read-write. Reset: 0. Enable PPR auto response when PPR is near overflow.
6:5	**PrivAbortEn**. Read-write. Reset: 0. Only effective when IOMMUx34[UsSup] == 1. Bits: Description 00b: Blind abort is not performed, PMR field from the request is used to check for required permission. 01b: Abort all guest acess to pages with U/S == 0. 10b: Future use if DTE enable bit is implemented. 11b: Reserved.

4	Reserved.
3:2	**DteSegEn**. Read-write. Reset: 0. DTE Segmentation Enable Bits — Description 00b — DTE is not segmented. Only use device table base 0 defined in IOMMUx00 and IOMMUx04. 01b — DTE is broken into 2 segments. Use device table base 0 defined in IOMMUx00 and IOMMUx04 and device table base 1 defined in IOMMUx[130,128,120,118,110,108,100] and IOMMUx[134,12C,124,11C,114,10C,104]. 10b — DTE is broken into 4 segments. Use device table base 0 defined in IOMMUx00 and IOMMUx04 and device table bases 1/2/3 defined in IOMMUx[130,128,120,118,110,108,100] and IOMMUx[134,12C,124,11C,114,10C,104]. 11b — DTE is broken into 8 segments. Use device table base 0 defined in IOMMUx00 and IOMMUx04 and all device table bases defined in IOMMUx[130,128,120,118,110,108,100] and IOMMUx[134,12C,124,11C,114,10C,104].
1:0	**EventQ.** Read-write. Reset: 0. Bits — Description 00b — Use default event log queue for event logger with autoswap disabled. 01b — Use event logBqueue for event logger with autoswap disabled. 10b — Autoswap queue between default event log and event log. Always start with default event log after reset. 11b — Unused.

IOMMUx20 Exclusion Range Base Low

Bits	Description
31:12	**ExclBase[31:12]: exclusion range base address bits[31:12]**. Read-write. Reset: 0. ExclBase[51:12] = {IOMMUx20[ExclBase[51:32]], ExclBase[31:12]}. Specifies the 4Kbyte-aligned base address of the exclusion range.
11:2	Reserved.
1	**ExAllow: exclusion allow**. Read-write. Reset: 0. 1=All accesses to the exclusion range are forwarded untranslated. 0=The EX bit in the device table entry specifies if accesses to the exclusion range are translated.
0	**ExEn: exclusion enable**. Read-write. Reset: 0. 1=The exclusion range is enabled.

IOMMUx24 Exclusion Range Base High

Bits	Description
31:20	Reserved.
19:0	**ExclBase[51:32]: exclusion range base address bits[51:32]**. See: IOMMUx20[ExclBase[31:12]].

IOMMUx28 Exclusion Range Limit Low

Bits	Description
31:12	**ExclLimit[31:12]: exclusion range limit address bits[31:12]**. Read-write. Reset: 0. ExclLimit[51:12] = {IOMMUx2C[ExclLimit[51:32]], ExclLimit[31:12]}. ExclLimit[51:12] specfies the 4Kbyte-aligned limit address of the exclusion range.
11:0	Reserved.

IOMMUx2C Exclusion Range Limit High

Bits	Description
31:20	Reserved.
19:0	**ExclLimitHi**. See: IOMMUx28[ExclLimit[31:12]].

IOMMUx30 Extended Feature Low

Bits	Description
31:30	Reserved.
29:28	**EventF**. Read-only. Reset: 2. Indicating Event logging functionality supported in hardware. Bits — Description 00b — Event log dual buffer not supported. 01b — Event log dual buffer supported without autoswap. 10b — Event log dual buffer supported with autoswap. 11b — Reserved.
27:26	**GaF**. Read-only. Reset: 0. Indicating GA logging functionality supported in hardware. Bits — Description 00b — GA log dual buffer not supported. 01b — GA log dual buffer supported without autoswap. 10b — GA log dual buffer supported with autoswap. 11b — Reserved.
25:24	**PprF**. Read-only. Reset: 2. Indicating PPR logging functionality supported in hardware. Bits — Description 00b — PPR log dual buffer not supported. 01b — PPR log dual buffer supported without autoswap. 10b — PPR log dual buffer supported with autoswap. 11b — Reserved.
23:21	**GamSup: General AVIC Mode Supported.** Read-only. Reset:0. Indicates the AVIC mode supported. Bits — Description 000b — Interrupt Remapping Only. No AVIC supported. 001b — AVIC supported. 111b-010b — Reserved.

20:18	**SmiFRC: SMI-Filter Register Count**. Read-only. Reset: 2. Indicates the number of SMI interrupt filter registers. SmiFRC must be 000b when SmiFSup == 00b. Bits — Description 000b — 1 SMI filter registers. 001b — 2 SMI filter registers. 010b — 4 SMI filter registers. 011b — 8 SMI filter registers. 100b — 16 SMI filter registers. 111b-101b — Reserved.
17:16	**SmiFSup: SMI Filter Supported**. Read-only. Reset: 0. Specifies that SMI interrupts may be filtered. Bits — Description 00b — SMI interrupts are always passed-through. 01b — SMI interrupts are filtered under the control of SmiFEn and the SMI-filter registers. 11b-10b — Reserved.
15:14	**GlxSup**. Read-only. Reset: 1. Bits — Description 00b — GLX in the DTE is ignored and the IOMMU performs only single-level guest CR3 lookups. This value is not meaningful when GtSup == 0. 01b — Two-level GCR3 base address table is supported in hardware. 10b — Three-level GCR3 base address table is supported in hardware for 20-bit PASID values. 11b — Reserved.
13:12	Reserved.
11:10	**HATS: host address translation size**. Read-only. Reset: 2. The maximum number of host address translation levels supported. This value is not meaningful when GtSup == 0. IOMMU behavior is undefined if Next Level in a page directory entry exceeds the limit set by HATS. See D0F2x80[HatsW]. Bits — Description 00b — 4 levels. 01b — 5 levels. 10b — 6 levels. 11b — Reserved.
9	**PcSup: performance counters supported**. Read-only. Reset: 1. 1=Performance counters are supported.
8	**HeSup: hardware error registers supported**. Read-only. Reset: 0. 0=Hardware error registers do not report error information. 1=Error information is reported in hardware error registers.
7	**GaSup: guest virtual APIC supported**. Read-only. Reset: 1. 1=Guest Virtual APIC supported.
6	**IaSup: INVALIDATE_IOMMU_ALL supported**. Read-only. Reset: 1. 1=The INVALIDATE_IOMMU_ALL command is supported.
5	Reserved.
4	**GtSup: guest translation supported**. Read-only. Reset: 1. 1=Guest address translation is supported. 0=Only nested address translation is supported. When GtSup == 0, the following values in the DTE must be zero: GV, GLX and GCR3 Table Root Pointer. See IOMMUx18[GtEn].
3	**NxSup: no execute supported**. Read-only. Reset:1. 1=No-execute protection is supported. 0=No-execute protection is not supported.

2	**XtSup: x2 apic supported**. Read-only. Reset: 0. 1=The interrupt remapping table is expanded to support x2APIC interrupt information. 0=x2APIC support is disabled.
1	**PprSup: peripheral page service request (PPR) supported**. Read-only. Reset: 1. 1=Indicates that IOMMU handles page service request events from peripherals, the IOMMU supports the page service request queue, and that the second IOMMU interrupt can be used to signal peripheral page service request events.
0	**PrefSup: prefetch support**. Read-only. Reset: 0. 1=Indicates that IOMMU will accept PREFETCH_IOMMU_PAGES commands.

IOMMUx34 Extended Feature High

Bits	Description
31:14	Reserved.
13	**GmcIommuBypassSup**. Read-only. Reset: 1h. 0=Indicate IOMMU supports advertising GMC whether to route memory traffic via GARLIC or ONION.
12	**BlockStopMarkSup**. Read-only. Reset: 1h. Indicate blocking stop mark message is supported.
11:10	**MarcNum**. Read-only. Reset: 1h. Indicate number of MARC triplet registers are supported. Bits — Description 00b — 0 sets of MARC triplet register supported. 01b — 4 sets of MARC triplet register supported. 10b — Reserved for 8 sets of MARC triplet register supported. 11b — Unused.
9	**PprAutoRespSup**. Read-only. Reset: 1h. Indicate PPR auto response generation is supported.
8	**PprOverflowEarlySup**. Read-only. Reset: 1h. Indicate PPR early overflow notification is supported.
7:6	**DTESeg**. Read-only. Reset: 3h. Bits — Description 00b — No DTE segmentation is supported. 01b — 2-way DTE segmentation is supported. 10b — 4-way DTE segmentation is supported. 11b — 8-way DTE segmentation is supported.
5	**UsSup**. Read-only. Reset: 1h. 0=Indicate privileged page protection is not supported. 1=Indicate privileged page protection is supported.
4	Reserved.
3:0	**PasMax: PASID maximum**. Read-only. Reset: Fh. This specifies the maximum PASID value supported. This field is not meaningful when IOMMUx30[GtSup] == 0. See D0F2x84[PasMaxW]. Bits — Description 2h-0h — Reserved. Fh-3h — 2^(PasMax+1)-1.

IOMMUx38 PPR Log Base Address Low

Bits	Description
31:12	**PprBase[31:12]: PPR log base address bits[31:12]**. Read-write. Reset: 0. PprBase[51:12] = {IOMMUx3C[PprBase[51:32]], PprBase[31:12]}. PprBase[51:12] specifies the 4Kbyte-aligned base address of the PPR log.
11:0	Reserved.

IOMMUx3C PPR Log Base Address High

Bits	Description
31:28	Reserved.
27:24	**PprLen: PPR log length**. Read-write. Reset: 8h. Specifies the length of the PPR log in power of two increments. Bits — Description 7h-0h — Reserved. Fh-8h — 2^PprLen entries (2^PprLen*16 bytes).
23:20	Reserved.
19:0	**PprBase[51:32]: PPR log base address bits[51:32]**. See: IOMMUx38[31:12].

IOMMUx40 Hardware Error Upper Low

Bits	Description
31:0	**FirstEvCode[31:0]: first event code bits[31:0]**. Read-write. Reset: 0. FirstEvCode[59:0] = {IOMMUx44[FirstEvCode[59:32]], FirstEvCode[31:0]}. IOMMUx44[EvCode] and FirstEvCode[59:0] specify the upper 64 bits of the most recent hardware error detected by the IOMMU.

IOMMUx44 Hardware Error Upper High

Bits	Description
31:28	**EvCode: event code**. Read-write. Reset: 0. Event code for the type of error logged.
27:0	**FirstEvCode[59:32]: first event code bits[59:32]**. See: IOMMUx40[FirstEvCode[31:0]].

IOMMUx48 Hardware Error Lower Low

Bits	Description
31:0	**SecondEvCode[31:0]: second event code bits[31:0]**. Read-write. Reset: 0. SecondEvCode[63:0] = {IOMMUx4C[SecondEvCode[63:32]], SecondEvCode[31:0]}. SecondEvCode[63:0] specifies the lower 64 bits of the most recent hardware error detected by IOMMU.

IOMMUx4C Hardware Error Lower High

Bits	Description
31:0	**SecondEvCode[63:32]: second event code bits[63:32]**. See: IOMMUx48[SecondEvCode[31:0]].

IOMMUx50 Hardware Error Status

Bits	Description
31:2	Reserved.
1	**HEO: hardware error overflow**. Read-write. Reset: 0. Defines the contents of the IOMMU hardware error registers as having beeing overwritten. 0=Not overwritten. 1=Contents overwritten by new information. HEO is only valid when HEV == 1.
0	**HEV: hardware error valid**. Read-write. Reset: 0. 1=Contents of the IOMMU hardware error registers are valid.

IOMMUx[78,70,68,60] SMI Filter Low

Table 201: Register Mapping for IOMMUx[78,70,68,60]

Register	Function
IOMMUx60	SMI Filter 0
IOMMUx68	SMI Filter 1
IOMMUx70	SMI Filter 2
IOMMUx78	SMI Filter 3

Bits	Description
31:18	Reserved.
17	**SMIFlock: SMI Filter Lock**. Read-write. Reset: 0.
16	**SMIDV: SMI Device Valid**. IF (SMIFlock == 1) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. The SMI filter is enabled and the device ID specified in SmiDID is valid for SMI.
15:0	**SMIDid: SMI Device ID**. IF (SMIFlock == 1) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Specifies the device ID for which SMIs are forwarded upstream.

IOMMUx[7C,74,6C,64] SMI Filter High

Table 202: Register Mapping for IOMMUx[7C,74,6C,64]

Register	Function
IOMMUx64	SMI Filter 0
IOMMUx6C	SMI Filter 1
IOMMUx74	SMI Filter 2
IOMMUx7C	SMI Filter 3

Bits	Description
31:0	Reserved.

IOMMUxE0 Guest APIC Log Base Address Low

Bits	Description
31:12	**GaLogBase[31:12]: guest APIC log base address bits[31:12]**. Read-write. Reset: 0. GaLogBase[51:12] = {IOMMUxE4[GaLogBase[51:32], GaLogBase[31:12]]}. GaLogBase[51:12] specifies the 4Kbyte-aligned base address of the guest virtual APIC log.
11:0	Reserved.

IOMMUxE4 Guest APIC Log Base Address High

Bits	Description
31:28	Reserved.
27:24	**GaLogLen: guest APIC log length**. Read-write. Reset: 8h. Indicates the lenght of the guest virtual APIC log.
23:20	Reserved.
19:0	**GaLogBase[51:32]: guest APIC log base address bit[51:32]**. See: IOMMUxE0[GaLogBase[31:12]].

IOMMUxE8 Guest APIC Log Tail Address Low

Bits	Description
31:3	**GaTAddr[31:3]: guest APIC log tail address[31:12]**. Read-write. Reset: 0.
2:0	Reserved.

IOMMUxEC Guest APIC Log Tail Address High

Bits	Description
31:20	Reserved.
19:0	**GaTAddr[51:32]: guest APIC log tail address [51:32]**. See: IOMMUxE8[GaTAddr[31:3]].

IOMMUxF0 PPR LogB Base Address Low

Bits	Description
31:12	**PprBBase[31:12]: PPR logB base address bits[31:12]**. Read-write. Reset: 0. PprBBase[51:12] = {IOMMUxF4[PprBBase[51:32]], PprBBase[31:12]]}. PprBBase[51:12] specifies the 4Kbyte-aligned base address of the PPR logB.
11:0	Reserved.

IOMMUxF4 PPR LogB Base Address High

Bits	Description
31:28	Reserved.
27:24	**PprBLen: PPR logB length**. Read-write. Reset: 8h. Indicates the length of the PPR logB in power of two increments. Bits — Description 7h-0h — Reserved. Fh-8h — 2^PprBLen entries (2^PprBLen*16 bytes).
23:20	Reserved.
19:0	**PprBBase[51:32]: PPR logB base address bit[51:32]**. See: IOMMUxF0[PprBBase[31:12]].

IOMMUxF8 Event LogB Base Address Low

Bits	Description
31:12	**EventBBase[31:12]: event logB base address bits[31:12]**. Read-write. Reset: 0. EventBBase[51:12] = {IOMMUxF8[EventBBase[51:32]], EventBBase[31:12]]}. EventBBase[51:12] specifies the 4Kbyte-aligned base address of the event logB.
11:0	Reserved.

IOMMUxFC Event LogB Base Address High

Bits	Description
31:28	Reserved.
27:24	**EventBLen: event logB length**. Read-write. Reset: 8h. Specifies the length of the event logB in power of 2 increments. The minimum size is 256 entries (4K bytes); values less than 8h are reserved. Bits — Description 7h-0h — Reserved. Fh-8h — 2^EventBLen entries (2^EventBLen*16 bytes).
23:20	Reserved.
19:0	**EventBBase[51:32]: event logB base address bit[51:32]**. See: IOMMUxF8[EventBBase[31:12]].

IOMMUx[130,128,120,118,110,108,100] Device Table Base Address Low

Table 203: Register Mapping for IOMMUx[130,128,120,118,110,108,100]

Register	Function	Register	Function
IOMMUx100	Device Table Base 1	IOMMUx120	Device Table Base 5
IOMMUx108	Device Table Base 2	IOMMUx128	Device Table Base 6
IOMMUx110	Device Table Base 3	IOMMUx130	Device Table Base 7
IOMMUx118	Device Table Base 4		

Bits	Description
31:12	**DevTblBase[31:12]: device table base address bits[31:12]**. Read-write. Reset: 0. DevTblBase[51:12] = {IOMMUx[134,12C,124,11C,114,10C,104][DevTblBase[51:32], DevTblBase[31:12]]}. DevTblBase[51:12] specifies the 4Kbyte-aligned base address of the first level device table.
11:9	Reserved.
8:0	**DevTblSize: device table size**. Read-write. Reset: 0. This field contains 1 less than the length of the device table, in multiples of 4K bytes. A minimum size of 0 corresponds to a 4K byte device table and a maximum size of 1FFh corresponds to a 2M byte device table.

IOMMUx[134,12C,124,11C,114,10C,104] Device Table Base Address High

Table 204: Register Mapping for IOMMUx[134,12C,124,11C,114,10C,104]

Register	Function	Register	Function
IOMMUx104	Device Table Base 1	IOMMUx124	Device Table Base 5
IOMMUx10C	Device Table Base 2	IOMMUx12C	Device Table Base 6
IOMMUx114	Device Table Base 3	IOMMUx134	Device Table Base 7
IOMMUx11C	Device Table Base 4		

Bits	Description
31:20	Reserved.
19:0	**DevTblBase[51:32]: device table base address bits[51:32]**. See: IOMMUx[130,128,120,118,110,108,100][DevTblBase[31:12]].

IOMMUx138 DSFX

Bits	Description
31:24	**RevID: revision ID**. Read-only. Reset: 0. See D0F2x8C[RevID].
23:0	**DsfxSup**. Read-only. Reset: 0. Indicate to software the feature that is supported. See D0F2x8C[DsfxSup].

IOMMUx140 DSCX

Bits	Description
31:24	**RevID: revision ID**. Read-only. Reset: 0. See D0F2x8C[RevID].
23:0	**DscxCntl**. Read-write. Reset:0. Indicate the bits to be used in hardware as controls for features. If the corresponding DSFX in IOMMUx138[DsfxSup] is low, the register bit should become read-only.

IOMMUx148 DSSX

Bits	Description
31:24	**RevID: revision ID**. Read-only. Reset: 0. See D0F2x8C[RevID].
23:0	**DssxStatus**. Read-write. Write-1-to-clear. Reset: 0. Reporting hardware status to software. See D0F2x90[DssxStatusSet].

IOMMUx150 Miscellenous Capabilities Low

Bits	Description
31:27	**PprMsiNum**. Read-only. Reset: 0. Indicates the MSI vector used for the interrupt message generated by the IOMMU for the PPR log when IOMMUx30[PprSup] == 1.
26:5	Reserved.
4:0	**MsiNum**. Read-only. Reset: 0. Indicates the MSI vector used for the interrupt message generated by the IOMMU.

IOMMUx154 Miscellenous Capabilities High

Bits	Description
31:5	Reserved.
4:0	**GaMsiNum**: Guest virtual APIC MSI Number. Read-only. Reset: 0. Indicates the MSI vector used for the interrupt message generated by the IOMMU for GA log.

IOMMUx158 MSI Capabilities

Bits	Description
31:24	Reserved.
23	**Msi64En**. Read-only. Reset: 1. Indicate 64-bit MSI address is supported.
22:20	**MsiMultiEn**. Read-write. Reset: 0. Sets the number of MSI messages assigned to this function.
19:17	**MsiMultiCap**. Read-only. Reset: 0. Indicate the number of MSI messages requested by this function.
16	**MsiEn**. Read-write. Reset: 0. Enable MSI and disable legacy interrupt for this function.
15:8	**MsiCapPtr**. Read-only. Reset: 74h. Pointer to the next configuration space capability.
7:0	**MsiCapId**. Read-only. Reset: 5h. Indication of a MSI capability.

IOMMUx15C MSI Address Low

Bits	Description
31:2	**MsiAddr[31:2]**. Read-write. Reset: 0. MsiAddr[63:2] = {IOMMUx160[MsiAddr[63:32]], MsiAddr[31:2]}. MsiAddr[63:2] specifies the address used to issue MSI messages.
1:0	Reserved.

IOMMUx160 MSI Address High

Bits	Description
31:0	**MsiAddr[63:32]**. See: IOMMUx15C[MsiAddr[31:2]].

IOMMUx164 MSI Data

Bits	Description
31:16	Reserved.
15:0	**MsiData**. Read-write. Reset: 0. Specifies the data issued with MSI messages.

IOMMUx168 MSI Mapping Capability

Bits	Description
31:27	**MsiMapType**. Read-only. Reset: 15h. Indicate MSI mapping capability.
26:18	**MsiMapRsv**. Read-only. Reset: 0. Reserved, control no hardware.
17	**MsiMapFix**. Read-only. Reset: 1. Indicate the device only maps MSI interrupts with address FEEX_XXXXh onto Hypertransport™ interrupts.
16	**MsiMapEn**. Read-only. Reset: 1. Indicate the MSI Mapping Capability is enabled.
15:8	**MsiMapCapPtr**. Read-only. Reset: 0. Point to the next capability list item.
7:0	**MsiMapCapId**. Read-only. Reset: 8h. Indicate a Hypertransport capability list item.

IOMMUx16C MSI Control

Bits	Description
31:14	Reserved.
13	**GmcIommuBypass**. Read-write. Reset: 0. To disable IOMMU from doing host translations or checks. IOMMU will indicate to GMC as IOMMU disabled and GMC should route memory traffic via GARLIC when this bit is set.
12:10	**MsiMultiCapW**. Read-write. Reset: 2h. This field sets the value of IOMMUx158[MsiMultiCap].
9:0	Reserved.

IOMMUx[248,230,218,200] MARC Base Address Low

Table 205: Register Mapping for IOMMUx[248,230,218,200]

Register	Function
IOMMUx200	MARC triplet 0
IOMMUx218	MARC triplet 1
IOMMUx230	MARC triplet 2
IOMMUx248	MARC triplet 3

Bits	Description
31:12	**MarcBaseAddr[31:12]**. Read-write. Reset: 00000h. MarcBaseAddr[51:12] = {IOMMUx[24C,234,21C,204][MarcBaseAddr[51:32]], MarcBaseAddr[31:12]}. MarcBaseAddr is ignored when IOMMUx[250,238,220,208][MarcEnable] == 0.
11:0	Reserved.

IOMMUx[24C,234,21C,204] MARC Base Address High

Table 206: Register Mapping for IOMMUx[24C,234,21C,204]

Register	Function
IOMMUx204	MARC triplet 0
IOMMUx21C	MARC triplet 1
IOMMUx234	MARC triplet 2
IOMMUx24C	MARC triplet 3

Bits	Description
31:20	Reserved.
19:0	**MarcBaseAddr[51:32]**. Read-write. Reset: 0. See IOMMUx[248,230,218,200][MarcBaseAddr[31:12]].

IOMMUx[250,238,220,208] MARC Relocation Address Low

Table 207: Register Mapping for IOMMUx[250,238,220,208]

Register	Function
IOMMUx208	MARC triplet 0
IOMMUx220	MARC triplet 1
IOMMUx238	MARC triplet 2
IOMMUx250	MARC triplet 3

Bits	Description
31:12	**MarcRelocAddr[31:12]**. Read-write. Reset: 0. MarcRelocAddr[51:12] = {IOMMUx[254,23C,224,20C][MarcRelocAddr[51:32]], MarcRelocAddr[31:12]}.
11:2	Reserved.
1	**MarcRO**. Read-write. Reset: 0. 0=Read and write accesses are processed when address matched. 1=Read accesses that match the address are routed via garlic and write accesses that match the address are routed via ONION.
0	**MarcEnable**. Read-write. Reset: 0. Enable routing controlled by the settings of the MARC triplet registers.

IOMMUx[254,23C,224,20C] MARC Relocatin Address High

Table 208: Register Mapping for IOMMUx[254,23C,224,20C]

Register	Function
IOMMUx20C	MARC triplet 0
IOMMUx224	MARC triplet 1
IOMMUx23C	MARC triplet 2
IOMMUx254	MARC triplet 3

Bits	Description
31:20	Reserved.
19:0	**MarcRelocAddr[51:32]**. Read-write. Reset: 0. See IOMMUx[250,238,220,208][MarcRelocAddr[31:12]].

IOMMUx[258,240,228,210] MARC Length Low

Table 209: Register Mapping for IOMMUx[258,240,228,210]

Register	Function
IOMMUx210	MARC triplet 0
IOMMUx228	MARC triplet 1
IOMMUx240	MARC triplet 2
IOMMUx258	MARC triplet 3

Bits	Description
31:12	**MarcLength[19:0]**. Read-write. Reset: 0. MarcLength[39:0] = {IOMMUx[25C,244,22C,214][MarcLength[39:20]], MarcLength[19:0]]}. The MARC length value is a multiple of 4KB. The value of 0 is reserved for future use.
11:0	Reserved.

IOMMUx[25C,244,22C,214] MARC Length High

Table 210: Register Mapping for IOMMUx[25C,244,22C,214]

Register	Function
IOMMUx214	MARC triplet 0
IOMMUx22C	MARC triplet 1
IOMMUx244	MARC triplet 2
IOMMUx25C	MARC triplet 3

Bits	Description
31:20	Reserved.
19:0	**MarcLength[39:20]**. Read-write. Reset: 0. See IOMMUx[258,240,228,210][MarcLength[19:0]].

IOMMUx1FF8 P2P Data

Bits	Description
31:0	**P2pData**. Read-write. Reset: 0.

IOMMUx2000 Command Buffer Head Pointer

Bits	Description
31:19	Reserved.
18:4	**CmdHdptr: command buffer head pointer**. Read-write; Updated-by-hardware. Reset: 0. Specifies the 128-bit aligned offset from the command buffer base address register of the next command to be fetched by the IOMMU. The IOMMU increments this register, rolling over to zero at the end of the buffer, after fetching and validating the command in the command buffer. After incrementing this register, the IOMMU cannot re-fetch the command from the buffer. If this register is written by software while IOMMUx2020[CmdBufRun] == 1, the IOMMU behavior is undefined. If this register is set by software to a value outside the length specified by IOMMUx0C[ComLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2008 Command Buffer Tail Pointer

Bits	Description
31:19	Reserved.
18:4	**CmdTailptr: command buffer tail pointer**. Read-write; Updated-by-hardware. Reset: 0. Specifies the 128-bit aligned offset from the command buffer base address register of the next command to be written by the software. Software must increment this field, rolling over to zero at the end of the buffer, after writing a command to the command buffer. If software advances the tail pointer equal to or beyond the head pointer after adding one or more commands to the buffer, the IOMMU behavior is undefined. If software sets the command buffer tail pointer to an offset beyond the length of the command buffer, the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2010 Event Log Head Pointer

Bits	Description
31:19	Reserved.
18:4	**EventHdptr: event log head pointer**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the event log base address register that will be read next by software. Software must increment this field, rolling over at the end of the buffer, after reading an event from the event log. If software advances the head pointer beyond the tail pointer, the IOMMU behavior is undefined. If software sets the event log head pointer to an offset beyond the length of the event log, the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2018 Event Log Tail Pointer

Bits	Description
31:19	Reserved.
18:4	**EventTailptr: event log tail pointer**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the event log base address register that will be written next by the IOMMU when an event is detected. The IOMMU increments this register, rolling over at the end of the buffer, after writing an event to the event log. If this register is written while IOMMUx2020[EventLogRun] == 1, the IOMMU behavior is undefined. If this register is set by software to a value outside the length specified by IOMMUx14[EventLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2020 Status

Bits	Description
31:19	Reserved.
18	**PprOverflowEarly**. Read-only; Write-1-to-clear. Reset: 0. 1=Default PPR log has reached early overflow threshold. See IOMMUx2088[PprOverflowEarlyThres]. When IOMMUx1C[PprAutoRespEn] == 1, auto responses will generate for the last page request and let stop marker message through the default PPR log.
17	**PprBOverflowEarly**. Read-only; Write-1-to-clear. Reset: 0. 1=PPR logB has reached early overflow threshold. See IOMMUx2090[PprBOverflowEarlyThres]. When IOMMUx1C[PprAutoRespEn] == 1, auto responses generate for the last page request and let stop marker message through the PPR logB.
16	**EventActive**. Read-only. Reset: 0. 0=Default event log being used by hardware. 1=Event logB being used by hardware.
15	**EventBOverflow**. Read-write; Write-1-to-clear. Reset: 0. 1=IOMMU event logB overflow has occurred. This bit is set when a new event is to be written to the event logB and there is no usable entry in the event logB, causing the new event information to be discarded. No new event logB entries are written while this bit is set. See IOMMUx18[EventIntEn].
14:13	Reserved.
12	**PprActive**. Read-only. Reset: 0. 0=Default PPR log being used by hardware. 1=PPR logB being used by hardware.
11	**PprBOverflow**. Read-write; Write-1-to-clear. Reset: 0. 1=IOMMU PPR logB overflow has occured. This bit is set when a new peripheral page service request is to be written to the PPR log and there is no usable entry in the PPR logB, causing the new information to be discarded. No new PPR logB entries are written while this bit is set. See IOMMUx18[PprIntEn].
10	**GaInt.** Read-write; Write-1-to-clear. Reset: 0. 1=GA request entry written to the GA log by the IOMMU. An interrupt is generated when this bit is set and IOMMUx18[GaIntEn] == 1. No new GA log entries are written when this bit is set.
9	**GaOverflow.** Read-write; Write-1-to-clear. Reset: 0. 1=IOMMU GA log overflow has occured. This bit is set when a new request is to be written to the GA log and there is no usable entry in the GA log, causing the new information to be discarded. No new GA log entries are written while this bit is set. See IOMMUx18[GaIntEn].

8	**GaRun.** Read-only. Reset: 0. 1=GA requests are logged as they occur. 0=GA requests are discarded without logging. When IOMMUx2020[GaOverflow] == 1, IOMMU does not write new GA log entries even when this bit is set. When halted, GA request logging is restarted by using IOMMUx18[GaLogEn].
7	**PprRun: peripheral page service request running**. Read-only. Reset: 0. 1=Peripheral page requests are logged as they occur. 0=Peripheral page requests are discarded without logging. When PprOverflow == 1, the IOMMU does not write new PPR log entries even when PprRun == 1. When halted, PPR logging is restarted by using IOMMUx18[PprLogEn].
6	**PprInt: peripheral page service request interrupt**. Read-write; Write-1-to-clear. Reset: 0. 1=PPR entry written to the PPR log by the IOMMU. 0=No PPR entry written to the PPR log by the IOMMU. See IOMMUx18[PprIntEn].
5	**PprOverflow: peripheral page service request overflow**. Read-write; Write-1-to-clear. Reset: 0. 1=IOMMU PPR log overflow has occured. This bit is set when a new peripheral page service request is to be written to the PPR log and there is no usable entry in the PPR log, causing the new information to be discarded. No new PPR log entries are written while this bit is set. See IOMMUx18[PprIntEn].
4	**CmdBufRun: command buffer running**. Read-only. Reset: 0. 1=Commands may be fetched from the command buffer. 0=IOMMU has stopped fetching new commands. The IOMMU freezes command processing after COMMAND_HARDWARE_ERROR or ILLEGAL_COMMAND_ERROR errors. When frozen, command fetching is restarted by using IOMMUx18[CmdBufEn].
3	**EventLogRun: event log running**. Read-only. Reset: 0. 1=Events are logged as they occur. 0=Event reports are discarded without logging. When EventOverflow == 1, the IOMMU does not write new event log entries even when EventLogRun == 1. When halted, event logging is restarted by using IOMMUx18[EventLogEn].
2	**ComWaitInt: completion wait interrupt**. Read-write; Write-1-to-clear. Reset: 0. 1=COMPLETION_WAIT command completed. This bit is only set if the i bit is set in the COMPLETION_WAIT command. See IOMMUx18[ComWaitIntEn].
1	**EventLogInt: event log interrupt**. Read-write; Write-1-to-clear. Reset: 0. 1=Event entry written to the event log by the IOMMU. See IOMMUx18[EventIntEn].
0	**EventOverflow**. Read-write; Write-1-to-clear. Reset: 0. 1=IOMMU event log overflow has occurred. This bit is set when a new event is to be written to the event log and there is no usable entry in the event log, causing the new event information to be discarded. No new event log entries are written while this bit is set. See IOMMUx18[EventIntEn].

IOMMUx2030 PPR Log Head Pointer

Bits	Description
31:19	Reserved.
18:4	**PprHdptr: PPR head pointer**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the PPR log base address register that will be read next by software. Software must increment this field, rolling over at the end of the buffer, after reading a PPR entry from the PPR event log. If software advances the head pointer beyond the tail pointer, the IOMMU behavior is undefined. If software sets the PPR log head pointer to an offset beyond the length of the PPR log, the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2038 PPR Log Tail Pointer

Bits	Description
31:19	Reserved.
18:4	**PprTailptr**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the PPR log base address register that will be written next by the IOMMU when a peripheral page request is detected. The IOMMU increments this register, rolling over at the end of the buffer, after writing a PPR entry to the PPR log. If this register is written while IOMMUx2020[PprRun] == 1, the IOMMU behavior is undefined. If software sets the PPR log tail pointer to an offset beyond the length of the PPR log, defined by IOMMUx3C[PprLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2040 Guest APIC Log Head Pointe Low

Bits	Description
31:16	Reserved.
15:3	**GaHdptr**. Read-write. Reset: 0.
2:0	Reserved.

IOMMUx2048 Guest APIC Log Tail Pointer Low

Bits	Description
31:16	Reserved.
15:3	**GaTailptr**. Read-write. Reset: 0.
2:0	Reserved.

IOMMUx2050 PPR LogB Head Pointer

Bits	Description
31:19	Reserved.
18:4	**PprBHdptr: PPR logB head pointer**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the PPR logB base address register that will be read next by software. Software must increment this field, rolling over at the end of the buffer, after reading a PPR entry from the PPR event logB. If software advances the head pointer beyond the tail pointer, the IOMMU behavior is undefined. If software sets the PPR logB head pointer to an offset beyond the length of the PPR logB, defined by IOMMUxF4[PprBLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2058 PPR LogB Tail Pointer

Bits	Description
31:19	Reserved.

18:4	**PprBTailptr**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the PPR logB base address register that is written next by the IOMMU when a peripheral page request is detected. The IOMMU increments this register, rolling over at the end of the buffer, after writing a PPR entry to the PPR logB. If this register is written while IOMMUx2020[PprRun] == 1, the IOMMU behavior is undefined. If software sets the PPR log tail pointer to an offset beyond the length of the PPR logB, defined by IOMMUxF4[PprBLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2070 Event LogB Head Pointer

Bits	Description
31:19	Reserved.
18:4	**EventBHdptr: event logB head pointer**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the event logB base address register that is read next by software. Software must increment this field, rolling over at the end of the buffer, after reading an event from the event logB. If software advances the head pointer beyond the tail pointer, the IOMMU behavior is undefined. If software sets the event logB head pointer to an offset beyond the length of the event logBspecified by IOMMUxFC[EventBLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2078 Event LogB Tail Pointer

Bits	Description
31:19	Reserved.
18:4	**EventBTailptr: event logB tail pointer**. Read-write. Reset: 0. Specifies the 128-bit aligned offset from the event logB base address register that is written next by the IOMMU when an event is detected. The IOMMU increments this register, rolling over at the end of the buffer, after writing an event to the event logB. If this register is written while IOMMUx2020[EventLogRun] == 1, the IOMMU behavior is undefined. If this register is set by software to a value outside the length specified by IOMMUxFC[EventBLen], the IOMMU behavior is undefined.
3:0	Reserved.

IOMMUx2080 PPR Auto Response

Bits	Description
31:5	Reserved.
4	**PprAutoRespMaskGN**. Read-write. Reset: 0h. 0=Auto Response forwards GN and PASID values from PPR. 1=Auto Response masks GN and PASID to 0 value.
3:0	**PprAutoRespCode**. Read-write. Reset: 0h. Programmable the response code use for auto response. Bits — Description 0h — Success. Fh-1h — Unused.

IOMMUx2088 PPR Log Overflow Early

Bits	Description
31	**PprOverflowEarlyEn**. Read-write. Reset: 0h. Enable PPR early overflow hardware mechanism on PPR log.
30	**PprOverflowEarlyIntEn**. Read-write. Reset: 0h. Enable PPR early overflow interrupt on PPR log.
29:15	Reserved.
14:0	**PprOverflowEarlyThres**. Read-write. Reset: 0h. Determine the number of entries prior to PPR log overflow that the hardware signals early overflow.

IOMMUx2090 PPR LogB Overflow Early

Bits	Description
31	**PprBOverflowEarlyEn**. Read-write. Reset: 0h. Enable PPR early overflow hardware mechanism on PPR logB.
30	**PprBOverflowEarlyIntEn**. Read-write. Reset: 0h. Enable PPR early overflow interrupt on PPR logB.
29:15	Reserved.
14:0	**PprBOverflowEarlyThres**. Read-write. Reset: 0h. Determine the number of entries prior to PPR logB overflow that the hardware signals early overflow.

IOMMUx4000 Counter Configuration

Bits	Description
31:18	Reserved.
17:12	**NCounterBanks: number of counter banks**. Read-only. Reset: 2h. The number of counter banks supported by the IOMMU. Each bank contains two or more counter and control registers as specified by NCounter. For each counter bank, a corresponding control bit is in IOMMUx4008, IOMMUx4010, and IOMMUx4018. Each supported event counter bank is in a distinct, consecutive 4K byte page. Bits — Description 00h — No counter banks supported. 3Fh-01h — <NCounterBanks> event counter banks are supported. Note: IOMMU event counter banks are numbered starting with 0.
11	Reserved.
10:7	**NCounter: number of counters per bank**. Read-only. Reset: 4h. Reports the number of individual counters in each IOMMU counter bank. Each counter bank contains the same number of counters. Bits — Description 0h — No counters supported. 1h — Reserved. Fh-2h — <NCounter> counters in each bank.
6:0	Reserved.

IOMMUx4008 Counter PASID Bank Lock Low

Bits	Description
31:0	**PasidLock[31:0]: pasid lock enable bits[31:0]**. Read-write. Reset: 0. PasidLock[63:0] = {IOMMUx400C[PasidLock[63:32]], PasidLock[31:0]}. For each bit in PasidLock[63:0], if the bit is set then writes to the corresponding bank in IOMMUx4[1,0][3:0]10 and IOMMUx4[1,0][3:0]14 are ignored. Bit positions above the value reported in IOMMUx4000[NCounterBanks] are ignored when written and return zero when read.

IOMMUx400C Counter PASID Bank Lock High

Bits	Description
31:0	**PasidLock[63:32]**. See: IOMMUx4008[PasidLock[31:0]].

IOMMUx4010 Domain Bank Lock Low

Bits	Description
31:0	**DomainLock[31:0]: domain lock enable bits[31:0]**. Read-write. Reset: 0. DomainLock[63:0] = {IOMMUx4014[DomainLock[63:32]], DomainLock[31:0]}. For each bit in DomainLock[63:0], if the bit is set then writes to the corresponding bank in IOMMUx4[1,0][3:0]18 and IOMMUx4[1,0][3:0]1C are ignored. Bit positions above the value reported in IOMMUx4000[NCounterBanks] are ignored when written and return zero when read.

IOMMUx4014 Domain Bank Lock High

Bits	Description
31:0	**DomainLock[63:32]**. See: IOMMUx4010[DomainLock[31:0]].

IOMMUx4018 DeviceID Bank Lock Low

Bits	Description
31:0	**DevIDLock[31:0]: deviceID lock enable bits[31:0]**. Read-write. Reset: 0. DevIDLock[63:0] = {IOMMUx401C[DevIDLock[63:32]], DevIDLock[31:0]}. For each bit in DevIDLock[63:0], if the bit is set then writes to the corresponding bank in IOMMUx4[1,0][3:0]20 and IOMMUx4[1,0][3:0]24 are ignored. Bit positions above the value reported in IOMMUx4000[NCounterBanks] are ignored when written and return zero when read.

IOMMUx401C DeviceID Bank Lock High

Bits	Description
31:0	**DevIDLock[63:32]**. See: IOMMUx4018[DevIDLock[31:0]].

IOMMUx4[1,0][3:0]00 Counter Low

Table 211: Register Mapping for IOMMUx4[1,0][3:0]00

Register	Function	Register	Function
IOMMUx40000	Bank 0 Counter 0	IOMMUx41000	Bank 1 Counter 0
IOMMUx40100	Bank 0 Counter 1	IOMMUx41100	Bank 1 Counter 1
IOMMUx40200	Bank 0 Counter 2	IOMMUx41200	Bank 1 Counter 2
IOMMUx40300	Bank 0 Counter 3	IOMMUx41300	Bank 1 Counter 3

Bits	Description
31:0	**Icounter[31:0]**. Read-write. Reset: 0. Icounter[47:0] = {IOMMUx4[1,0][3:0]04[Icounter[47:32]], Icounter[31:0]}. Icounter[47:0] reports the counter value. The counter counts up continuously, wrapping at the maximum value. There is no overflow indicator.

IOMMUx4[1,0][3:0]04 Counter High

Table 212: Register Mapping for IOMMUx4[1,0][3:0]04

Register	Function	Register	Function
IOMMUx40004	Bank 0 Counter 0	IOMMUx41004	Bank 1 Counter 0
IOMMUx40104	Bank 0 Counter 1	IOMMUx41104	Bank 1 Counter 1
IOMMUx40204	Bank 0 Counter 2	IOMMUx41204	Bank 1 Counter 2
IOMMUx40304	Bank 0 Counter 3	IOMMUx41304	Bank 1 Counter 3

Bits	Description
31:16	Reserved.
15:0	**Icounter[47:32]**. See: IOMMUx4[1,0][3:0]00[Icounter[31:0]].

IOMMUx4[1,0][3:0]08 Counter Source

Table 213: Register Mapping for IOMMUx4[1,0][3:0]08

Register	Function	Register	Function
IOMMUx40008	Bank 0 Counter 0	IOMMUx41008	Bank 1 Counter 0
IOMMUx40108	Bank 0 Counter 1	IOMMUx41108	Bank 1 Counter 1
IOMMUx40208	Bank 0 Counter 2	IOMMUx41208	Bank 1 Counter 2
IOMMUx40308	Bank 0 Counter 3	IOMMUx41308	Bank 1 Counter 3

Bits	Description
31	**Cac: counter source architectural or custom**. Read-write. Reset: 0. 0=Architectural counter input group. 1=Custom input group.
30	**CountUnits**. Read-write. Reset: 0. 0=Counter counts events (level). 1=Counter counts clocks (edges).

29:8	Reserved.
7:0	**Csource: counter source**. Read-write. Reset: 0. Counter source. Selects event counter input from the choices provided.

IOMMUx4[1,0][3:0]10 PASID Match Low

See IOMMUx4008.

Table 214: Register Mapping for IOMMUx4[1,0][3:0]10

Register	Function	Register	Function
IOMMUx40010	Bank 0 Counter 0	IOMMUx41010	Bank 1 Counter 0
IOMMUx40110	Bank 0 Counter 1	IOMMUx41110	Bank 1 Counter 1
IOMMUx40210	Bank 0 Counter 2	IOMMUx41210	Bank 1 Counter 2
IOMMUx40310	Bank 0 Counter 3	IOMMUx41310	Bank 1 Counter 3

Bits	Description
31	**PasMEn: PASID match enable**. Read-write. Reset: 0. 0=PASID is ignored. 1=Filtered PASID must match to count an event. An event with no PASID tag is only counted when Pasmen == 0.
30:16	Reserved.
15:0	**PasidMatch**. Read-write. Reset: 0. This value is compared with the masked (filtered) value of the incoming PASID of the transaction to decide if the corresponding event is counted. The event is counted if PasidMatch is exactly equal to the masked incoming PASID.

IOMMUx4[1,0][3:0]14 PASID Match High

See IOMMUx4008.

Table 215: Register Mapping for IOMMUx4[1,0][3:0]14

Register	Function	Register	Function
IOMMUx40014	Bank 0 Counter 0	IOMMUx41014	Bank 1 Counter 0
IOMMUx40114	Bank 0 Counter 1	IOMMUx41114	Bank 1 Counter 1
IOMMUx40214	Bank 0 Counter 2	IOMMUx41214	Bank 1 Counter 2
IOMMUx40314	Bank 0 Counter 3	IOMMUx41314	Bank 1 Counter 3

Bits	Description
31:16	Reserved.
15:0	**PasidMask**. Read-write. Reset: 0. This bit-mask is ANDed with the PASID of the transaction to decide to count the corresponding event. 0000h=Count events for all values of incoming PASID. 0001h-FFFFh=Bit-wise mask ANDed with incoming PASID.

IOMMUx4[1,0][3:0]18 Domain Match Low

Table 216: Register Mapping for IOMMUx4[1,0][3:0]18

Register	Function	Register	Function
IOMMUx40018	Bank 0 Counter 0	IOMMUx41018	Bank 1 Counter 0
IOMMUx40118	Bank 0 Counter 1	IOMMUx41118	Bank 1 Counter 1
IOMMUx40218	Bank 0 Counter 2	IOMMUx41218	Bank 1 Counter 2
IOMMUx40318	Bank 0 Counter 3	IOMMUx41318	Bank 1 Counter 3

Bits	Description
31	**DomMEn: domain match enable**. Read-write. Reset: 0. 0=Domain is ignored. 1=Filtered Domain must match DomainMatch to count an event.
30:16	Reserved.
15:0	**DomainMatch**. Read-write. Reset: 0. This value is compared with the masked (filtered) value of the incoming domain of the transaction to decide to count the corresponding event. The event is counted if DomainMatch is exactly equal to the masked incoming Domain.

IOMMUx4[1,0][3:0]1C Domain Match High

Table 217: Register Mapping for IOMMUx4[1,0][3:0]1C

Register	Function	Register	Function
IOMMUx4001C	Bank 0 Counter 0	IOMMUx4101C	Bank 1 Counter 0
IOMMUx4011C	Bank 0 Counter 1	IOMMUx4111C	Bank 1 Counter 1
IOMMUx4021C	Bank 0 Counter 2	IOMMUx4121C	Bank 1 Counter 2
IOMMUx4031C	Bank 0 Counter 3	IOMMUx4131C	Bank 1 Counter 3

Bits	Description
31:16	Reserved.
15:0	**DomainMask**. Read-write. Reset: 0. This bit-mask is ANDed with the Domain of the transaction to decide to count the corresponding event. 0000h=Count events for all values of incoming Domain. 0001h-FFFFh=Bit-wise mask ANDed with incoming Domain.

IOMMUx4[1,0][3:0]20 DeviceID Match Low

Table 218: Register Mapping for IOMMUx4[1,0][3:0]20

Register	Function	Register	Function
IOMMUx40020	Bank 0 Counter 0	IOMMUx41020	Bank 1 Counter 0
IOMMUx40120	Bank 0 Counter 1	IOMMUx41120	Bank 1 Counter 1
IOMMUx40220	Bank 0 Counter 2	IOMMUx41220	Bank 1 Counter 2
IOMMUx40320	Bank 0 Counter 3	IOMMUx41320	Bank 1 Counter 3

Bits	Description
31	**DidMEn: deviceID match enable**. Read-write. Reset: 0. 0=DeviceID is ignored. 1=Filtered DeviceID must match to count an event.
30:16	Reserved.
15:0	**DeviceidMatch**. Read-write. Reset: 0. This value is compared with the masked (filtered) value of the incoming DeviceID of the transaction to decide to count the corresponding event. The event is counted if DeviceidMatch is exactly equal to the masked incoming DeviceID.

IOMMUx4[1,0][3:0]24 DeviceID Match High

Table 219: Register Mapping for IOMMUx4[1,0][3:0]24

Register	Function	Register	Function
IOMMUx40024	Bank 0 Counter 0	IOMMUx41024	Bank 1 Counter 0
IOMMUx40124	Bank 0 Counter 1	IOMMUx41124	Bank 1 Counter 1
IOMMUx40224	Bank 0 Counter 2	IOMMUx41224	Bank 1 Counter 2
IOMMUx40324	Bank 0 Counter 3	IOMMUx41324	Bank 1 Counter 3

Bits	Description
31:16	Reserved.
15:0	**DeviceidMask**. Read-write. Reset: 0. This bit-mask is ANDed with the DeviceID of the transaction to decide to count the corresponding event. 0000h=Count events for all values of incoming DeviceID. 0001h-FFFFh=Bit-wise mask ANDed with incoming DeviceID.

IOMMUx4[1,0][3:0]28 Counter Report Low

Table 220: Register Mapping for IOMMUx4[1,0][3:0]28

Register	Function	Register	Function
IOMMUx40028	Bank 0 Counter 0	IOMMUx41028	Bank 1 Counter 0
IOMMUx40128	Bank 0 Counter 1	IOMMUx41128	Bank 1 Counter 1
IOMMUx40228	Bank 0 Counter 2	IOMMUx41228	Bank 1 Counter 2
IOMMUx40328	Bank 0 Counter 3	IOMMUx41328	Bank 1 Counter 3

Bits	Description
31:0	**EventNote[31:0]**. Read-write. Reset: 0. EventNote[51:0] = {IOMMUx4[1,0][3:0]2C[EventNote[51:32]], EventNote[31:0]}. When IOMMUx4[1,0][3:0]2C[CERE] == 1 and the corresponding counter is incremented and wraps to zero, EventNote[51:0] is reported in the EVENT_COUNTER_ZERO event log entry.

IOMMUx4[1,0][3:0]2C Counter Report High

Table 221: Register Mapping for IOMMUx4[1,0][3:0]2C

Register	Function	Register	Function
IOMMUx4002C	Bank 0 Counter 0	IOMMUx4102C	Bank 1 Counter 0
IOMMUx4012C	Bank 0 Counter 1	IOMMUx4112C	Bank 1 Counter 1
IOMMUx4022C	Bank 0 Counter 2	IOMMUx4122C	Bank 1 Counter 2
IOMMUx4032C	Bank 0 Counter 3	IOMMUx4132C	Bank 1 Counter 3

Bits	Description
31	**CERE: counter event report enable**. Read-write. Reset: 0. Counter Event Report Enable. 0=No event report when counter wraps to zero. 1=IOMMU writes an EVENT_COUNTER_ZERO event log entry when the counter wraps to zero. The counter-wrap event is treated like any other event. Software note: the counter-wrap event is delivered promptly but without a latency assurance.
30:20	Reserved.
19:0	**EventNote[51:32]**. See: IOMMUx4[1,0][3:0]28[EventNote[31:0]].

3.18 APIC Registers

See 2.4.9.1.2 [APIC Register Space].

MMIO local APIC space is accessible in xAPIC mode.

APIC20 APIC ID

Bits	Description
31:24	**ApicId: APIC ID**. Read-write. Reset: Varies based on core number. • The initial value of APIC20[ApicId[7:0]] is {0000b, CpuCoreNum[3:0]}. See 2.4.9.1.3 [ApicId Enumeration Requirements]. See 2.4.4 [Processor Cores and Downcoring].
23:0	Reserved.

APIC30 APIC Version

Bits	Description
31	**ExtApicSpace: extended APIC register space present**. Read-only. Reset: 1. 1=Indicates the presence of extended APIC register space starting at APIC400.
30:25	RAZ.
24	**DirectedEoiSupport: directed EOI support**. Read-only. Reset: 0. 0=Directed EOI capability not supported.
23:16	**MaxLvtEntry**. Read-only. Reset: Product-specific. Specifies the number of entries in the local vector table minus one.
15:8	RAZ.
7:0	**Version**. Read-only. Reset: 10h. Indicates the version number of this APIC implementation.

APIC80 Task Priority (TPR)

Bits	Description
31:8	RAZ.
7:0	**Priority**. Read-write. Reset: 0. This field is assigned by software to set a threshold priority at which the core is interrupted.

APIC90 Arbitration Priority (APR)

Bits	Description
31:8	RAZ.
7:0	**Priority**. Read-only. Reset: 0. Indicates the current priority for a pending interrupt, or a task or interrupt being serviced by the core. The priority is used to arbitrate between cores to determine which accepts a lowest-priority interrupt request.

APICA0 Processor Priority (PPR)

Bits	Description
31:8	RAZ.
7:0	**Priority**. Reset: 0. Read-only. Indicates the core's current priority servicing a task or interrupt, and is used to determine if any pending interrupts should be serviced. It is the higher value of the task priority value and the current highest in-service interrupt.

APICB0 End of Interrupt

This register is written by the software interrupt handler to indicate the servicing of the current interrupt is complete.

Bits	Description
31:0	Unused. Write-only.

APICC0 Remote Read

Bits	Description
31:0	**RemoteReadData**. Read-only. Reset: 0. The data resulting from a valid completion of a remote read inter-processor interrupt.

APICD0 Logical Destination (LDR)

Bits	Description
31:24	**Destination**. Read-write. Reset: 0. This APIC's destination identification. Used to determine which interrupts should be accepted.
23:0	Reserved.

APICE0 Destination Format

Only supported in xAPIC mode.

Bits	Description
31:28	**Format**. Read-write. Reset: Fh. Controls which format to use when accepting interrupts with a logical destination mode. Bits — Definition 0h — Cluster destinations are used Eh-1h — Reserved. Fh — Flat destinations are used
27:0	Reserved. Reset: FFF_FFFFh.

APICF0 Spurious-Interrupt Vector (SVR)

Bits	Description
31:13	RAZ.
12	**EoiBroadcastDisable: EOI broadcast disable**. Read-only. Reset: 0.
11:10	RAZ.
9	**FocusDisable**. Read-write. Reset: 0. 1=Disable focus core checking during lowest-priority arbitrated interrupts.
8	**APICSWEn: APIC software enable**. Read-write. Reset: 0. 0=SMI, NMI, INIT, LINT[1:0], and Startup interrupts may be accepted; pending interrupts in APIC[170:100] and APIC[270:200] are held, but further fixed, lowest-priority, and ExtInt interrupts are not accepted. All LVT entry mask bits are set and cannot be cleared.
7:0	**Vector**. Read-write. Reset: FFh. The vector that is sent to the core in the event of a spurious interrupt. The behavior of bits[3:0] are controlled as specified by D18F0x68[ApicExtSpur].

APIC[170:100] In-Service (ISR)

The in-service registers provide a bit per interrupt to indicate that the corresponding interrupt is being serviced by the core. APIC100[15:0] are reserved. Interrupts are mapped as follows:

Table 222: Register Mapping for APIC[170:100]

Register	Function
APIC100	Interrupts [31:16]
APIC110	Interrupts [63:32]
APIC120	Interrupts [95:64]
APIC130	Interrupts [127:96]
APIC140	Interrupts [159:128]
APIC150	Interrupts [191:160]
APIC160	Interrupts [223:192]
APIC170	Interrupts [255:224]

Bits	Description
31:0	**InServiceBits**. Reset: 0. Read-only. These bits are set when the corresponding interrupt is being serviced by the core.

APIC[1F0:180] Trigger Mode (TMR)

The trigger mode registers provide a bit per interrupt to indicate the assertion mode of each interrupt. APIC180[15:0] are reserved. Interrupts are mapped as follows:

Table 223: Register Mapping for APIC[1F0:180]

Register	Function
APIC180	Interrupts [31:16]
APIC190	Interrupts [63:32]

Table 223: Register Mapping for APIC[1F0:180] (Continued)

APIC1A0	Interrupts [95:64]
APIC1B0	Interrupts [127:96]
APIC1C0	Interrupts [159:128]
APIC1D0	Interrupts [191:160]
APIC1E0	Interrupts [223:192]
APIC1F0	Interrupts [255:224]

Bits	Description
31:0	**TriggerModeBits**. Reset: 0. Read-only. The corresponding trigger mode bit is updated when an interrupt is accepted. The values are: 0=Edge-triggered interrupt. 1=Level-triggered interrupt.

APIC[270:200] Interrupt Request (IRR)

The interrupt request registers provide a bit per interrupt to indicate that the corresponding interrupt has been accepted by the APIC. APIC200[15:0] are reserved. Interrupts are mapped as follows:

Table 224: Register Mapping for APIC[270:200]

Register	Function
APIC200	Interrupts [31:16]
APIC210	Interrupts [63:32]
APIC220	Interrupts [95:64]
APIC230	Interrupts [127:96]
APIC240	Interrupts [159:128]
APIC250	Interrupts [191:160]
APIC260	Interrupts [223:192]
APIC270	Interrupts [255:224]

Bits	Description
31:0	**RequestBits**. Read-only. Reset: 0. The corresponding request bit is set when the an interrupt is accepted by the APIC.

APIC280 Error Status

Writes to this register trigger an update of the register state. The value written by software is arbitrary. Each write causes the internal error state to be loaded into this register, clearing the internal error state. Consequently, a second write prior to the occurrence of another error causes the register to be overwritten with cleared data.

Bits	Description
31:8	RAZ.
7	**IllegalRegAddr: illegal register address**. Read-write. Reset: 0. This bit indicates that an access to a nonexistent register location within this APIC was attempted. Can only be set in xAPIC mode.

6	**RcvdIllegalVector: received illegal vector**. Read-write. Reset: 0. This bit indicates that this APIC has received a message with an illegal vector (00h to 0Fh for fixed and lowest priority interrupts).
5	**SentIllegalVector**. Read-write. Reset: 0. This bit indicates that this APIC attempted to send a message with an illegal vector (00h to 0Fh for fixed and lowest priority interrupts).
4	RAZ.
3	**RcvAcceptError: receive accept error**. Read-write. Reset: 0. This bit indicates that a message received by this APIC was not accepted by this or any other APIC.
2	**SendAcceptError**. Read-write. Reset: 0. This bit indicates that a message sent by this APIC was not accepted by any APIC.
1:0	RAZ.

APIC300 Interrupt Command Low (ICR Low)

Not all combinations of ICR fields are valid. Only the following combinations are valid:

Table 225: ICR valid combinations

Message Type	Trigger Mode	Level	Destination Shorthand
Fixed	Edge	x	x
	Level	Assert	x
Lowest Priority, SMI, NMI, INIT	Edge	x	Destination or all excluding self.
	Level	Assert	Destination or all excluding self
Startup	x	x	Destination or all excluding self

Note: x indicates a don't care.

Bits	Description
31:20	RAZ.
19:18	**DestShrthnd: destination shorthand**. Read-write. Reset: 0. Provides a quick way to specify a destination for a message. Bits — Description 00b — No shorthand (Destination field) 01b — Self 10b — All including self 11b — All excluding self (This sends a message with a destination encoding of all 1s, so if lowest priority is used the message could end up being reflected back to this APIC.) If all including self or all excluding self is used, then destination mode is ignored and physical is automatically used.

17:16	**RemoteRdStat: remote read status**. Read-only. Reset: 0. Bits — Description 00b — Read was invalid 01b — Delivery pending 10b — Delivery complete and access was valid 11b — Reserved
15	**TM: trigger mode**. Read-write. Reset: 0. Indicates how this interrupt is triggered. 0=Edge triggered. 1=Level triggered.
14	**Level**. Read-write. Reset: 0. 0=Deasserted. 1=Asserted.
13	RAZ.
12	**DS: interrupt delivery status**. Read-only. Reset: 0. This bit is set to indicate that the interrupt has not yet been accepted by the destination core(s). 0=Idle. 1=Send pending. Software may repeatedly write ICRL without polling the DS bit; all requested IPIs will be delivered.
11	**DM: destination mode**. Read-write. Reset: 0. 0=Physical. 1=Logical.
10:8	**MsgType**. Read-write. Reset: 0. The message types are encoded as follows: Bits — Description 000b — Fixed 001b — Lowest Priority 010b — SMI 011b — Remote read 100b — NMI 101b — INIT 110b — Startup 111b — External interrupt
7:0	**Vector**. Read-write. Reset: 0. The vector that is sent for this interrupt source.

APIC310 Interrupt Command High (ICR High)

Bits	Description
31:24	**DestinationField**. Read-write. Reset: 0. The destination encoding used when APIC300[DestShrthnd] is 00b.
23:0	RAZ.

APIC320 LVT Timer

Bits	Description
31:18	RAZ.
17	**Mode**. Read-write. Reset: 0. 0=One-shot. 1=Periodic.
16	**Mask**. Read-write. Reset: 1. 0=Not masked. 1=Masked.
15:13	RAZ.
12	**DS: interrupt delivery status**. Read-only; Updated-by-hardware. Reset: 0. 0=Idle. 1=Send pending. (Indicates that the interrupt has not yet been accepted by the core.)
11	RAZ.

10:8	**MsgType: message type**. Read-write. Reset: 000b. See 2.4.9.1.14 [Generalized Local Vector Table].
7:0	**Vector**. Read-write. Reset: 00h. Interrupt vector number.

APIC330 LVT Thermal Sensor

Interrupts for this local vector table are caused by changes in MSRC001_0061[CurPstateLimit] due to SB-RMI or HTC.

Bits	Description
31:17	RAZ.
16	**Mask**. Read-write. Reset: 1. 0=Not masked. 1=Masked.
15:13	RAZ.
12	**DS: interrupt delivery status**. Read-only; Updated-by-hardware. Reset: 0. 0=Idle. 1=Send pending. (Indicates that the interrupt has not yet been accepted by the core.)
11	RAZ.
10:8	**MsgType: message type**. Read-write. Reset: 000b. See 2.4.9.1.14 [Generalized Local Vector Table].
7:0	**Vector**. Read-write. Reset: 00h. Interrupt vector number.

APIC340 LVT Performance Monitor

Interrupts for this local vector table are caused by overflows of:

- MSRC001_00[07:04] [Performance Event Counter (PERF_CTR[3:0])].
- MSRC001_020[B,9,7,5,3,1] [Performance Event Counter (PERF_CTR[5:0])].
- MSRC001_024[7,5,3,1] [Northbridge Performance Event Counter (NB_PERF_CTR[3:0])].

Bits	Description
31:17	RAZ.
16	**Mask**. Read-write. Reset: 1. 0=Not masked. 1=Masked.
15:13	RAZ.
12	**DS: interrupt delivery status**. Read-only; Updated-by-hardware. Reset: 0. 0=Idle. 1=Send pending. (Indicates that the interrupt has not yet been accepted by the core.)
11	RAZ.
10:8	**MsgType: message type**. Read-write. Reset: 000b. See 2.4.9.1.14 [Generalized Local Vector Table].
7:0	**Vector**. Read-write. Reset: 00h. Interrupt vector number.

APIC3[60:50] LVT LINT[1:0]

Table 226: Register Mapping for APIC3[60:50]

Register	Function
APIC350	LINT 0
APIC360	LINT 1

Bits	Description
31:17	RAZ.
16	**Mask**. Read-write. Reset: 1. 0=Not masked. 1=Masked.
15	**TM: trigger mode**. Read-write. Reset: 0. 0=Edge. 1=Level.
14	**RmtIRR**. Read-only; Updated-by-hardware. Reset: 0. If trigger mode is level, remote IRR is set when the interrupt has begun service. Remote IRR is cleared when the end of interrupt has occurred.
13	Reserved.
12	**DS: interrupt delivery status**. Read-only; Updated-by-hardware. Reset: 0. 0=Idle. 1=Send pending. (Indicates that the interrupt has not yet been accepted by the core.)
11	RAZ.
10:8	**MsgType: message type**. Read-write. Reset: 000b. See 2.4.9.1.14 [Generalized Local Vector Table].
7:0	**Vector**. Read-write. Reset: 00h. Interrupt vector number.

APIC370 LVT Error

Bits	Description
31:17	RAZ.
16	**Mask**. Read-write. Reset: 1. 0=Not masked. 1=Masked.
15:13	Reserved.
12	**DS: interrupt delivery status**. Read-only; Updated-by-hardware. Reset: 0. 0=Idle. 1=Send pending. (Indicates that the interrupt has not yet been accepted by the core.)
11	RAZ.
10:8	**MsgType: message type**. Read-write. Reset: 000b. See 2.4.9.1.14 [Generalized Local Vector Table].
7:0	**Vector**. Read-write. Reset: 00h. Interrupt vector number.

APIC380 Timer Initial Count

Bits	Description
31:0	**Count**. Read-write. Reset: 0. The value copied into the current count register when the timer is loaded or reloaded.

APIC390 Timer Current Count

Bits	Description
31:0	**Count**. Read-only. Reset: 0. The current value of the counter.

APIC3E0 Timer Divide Configuration

The Div bits are encoded as follows:

Table 227: Div[3,1:0] Value Table

Div[3]	Div[1:0]	Resulting Timer Divide
0	00b	2
0	01b	4
0	10b	8
0	11b	16
1	00b	32
1	01b	64
1	10b	128
1	11b	1

Bits	Description
31:4	RAZ.
3	**Div[3]**. Read-write. Reset: 0. See Table 227.
2	RAZ.
1:0	**Div[1:0]**. Read-write. Reset: 0. See Table 227.

APIC400 Extended APIC Feature

Bits	Description
31:24	RAZ.
23:16	**ExtLvtCount: extended local vector table count**. Read-only. Reset: 04h. This specifies the number of extended LVT registers (APIC[530:500]) in the local APIC.
15:3	RAZ.
2	**ExtApicIdCap: extended APIC ID capable**. Read-only. Reset: 1. 1=The processor is capable of supporting an 8-bit APIC ID, as controlled by APIC410[ExtApicIdEn].
1	**SeoiCap: specific end of interrupt capable**. Read-only. Reset: 1. 1=The APIC420 [Specific End Of Interrupt] is present.
0	**IerCap: interrupt enable register capable**. Read-only. Reset: 1. This bit indicates that the APIC[4F0:480] [Interrupt Enable] are present. See 2.4.9.1.8 [Interrupt Masking].

APIC410 Extended APIC Control

Bits	Description
31:3	RAZ.

2	**ExtApicIdEn: extended APIC ID enable**. Read-write. Reset: 0. 1=Enable 8-bit APIC ID; APIC20[ApicId] supports an 8-bit value; an interrupt broadcast in physical destination mode requires that the IntDest[7:0]=1111_1111b (instead of XXXX_1111b); a match in physical destination mode occurs when (IntDest[7:0] == ApicId[7:0]) instead of (IntDest[3:0] == ApicId[3:0]). If ExtApicIdEn == 1 then program D18F0x68[ApicExtId] = 1 and D18F0x68[ApicExtBrdCst] = 1.
1	**SeoiEn**. Read-write. Reset: 0. 1=Enable SEOI generation when a write to APIC420 [Specific End Of Interrupt] is received.
0	**IerEn**. Read-write. Reset: 0. 1=Enable writes to the interrupt enable registers.

APIC420 Specific End Of Interrupt

Bits	Description
31:8	RAZ.
7:0	**EoiVec: end of interrupt vector**. Read-write. Reset: 0. A write to this field causes an end of interrupt cycle to be performed for the vector specified in this field. The behavior is undefined if no interrupt is pending for the specified interrupt vector.

APIC[4F0:480] Interrupt Enable

Interrupt enables range is mapped as follows:

Table 228: Register Mapping for APIC[4F0:480]

Register	Function
APIC480	IntEn[31:0]
APIC490	IntEn[63:32]
APIC4A0	IntEn[95:64]
APIC4B0	IntEn[127:96]
APIC4C0	IntEn[159:128]
APIC4D0	IntEn[191:160]
APIC4E0	IntEn[223:192]
APIC4F0	IntEn[255:224]

Bits	Description
31:0	**InterruptEnableBits**. Read-write. Reset: FFFF_FFFFh. The interrupt enable bits can be used to enable each of the 256 interrupts. See above table.

APIC[530:500] Extended Interrupt [3:0] Local Vector Table

APIC500 provides a local vector table entry for IBS; See D18F3x1CC. APIC510 provides a local vector table entry for error thresholding. APIC520 is used for the Deferred Error interrupt, and APIC530 is unused.

Table 229: Register Mapping for APIC[530:500]

Register	Function
APIC500	Extended Interrupt 0 (IBS)
APIC510	Extended Interrupt 1 (Thresholding)
APIC530	Extended Interrupt 3

Bits	Description
31:17	RAZ.
16	**Mask**. Read-write. Reset: 1. 0=Not masked. 1=Masked.
15:13	RAZ.
12	**DS: interrupt delivery status**. Read-only; Updated-by-hardware. Reset: 0. 0=Idle. 1=Send pending. (Indicates that the interrupt has not yet been accepted by the core.)
11	RAZ.
10:8	**MsgType: message type**. Read-write. Reset: 000b. See 2.4.9.1.14 [Generalized Local Vector Table].
7:0	**Vector**. Read-write. Reset: 00h. Interrupt vector number.

3.19 CPUID Instruction Registers

Processor feature capabilities and configuration information are provided through the CPUID instruction. The information is accessed by (1) selecting the CPUID function setting EAX and optionally ECX for some functions, (2) executing the CPUID instruction, and (3) reading the results in the EAX, EBX, ECX, and EDX registers. The syntax *CPUID FnXXXX_XXXX_EiX[_xYYY]* refers to the function where EAX == X, and optionally ECX == Y, and the registers specified by EiX. EiX can be any single register such as {EAX, EBX, ECX, and EDX}, or a range of registers, such as E[C,B,A]X. Undefined function numbers return 0's in all 4 registers. See 2.4.11 [CPUID Instruction].

Unless otherwise specified, single-bit feature fields are encoded as 1=Feature is supported by the processor; 0=Feature is not supported by the processor.

The following provides processor specific details about CPUID.

CPUID Fn0000_0000_EAX Processor Vendor and Largest Standard Function Number

Bits	Description
31:0	**LFuncStd: largest standard function**. Value: 0000_000Dh. The largest CPUID standard function input value supported by the processor implementation.

CPUID Fn0000_0000_E[D,C,B]X Processor Vendor

CPUID Fn0000_0000_E[D,C,B]X and CPUID Fn8000_0000_E[D,C,B]X return the same value.

Table 230: Reset Mapping for CPUID Fn8000_0000_E[D,C,B]X

Register	Value	Description
CPUID Fn0000_0000_EBX	6874_7541h	The ASCII characters “h t u A”.
CPUID Fn0000_0000_ECX	444D_4163h	The ASCII characters “D M A c”.
CPUID Fn0000_0000_EDX	6974_6E65h	The ASCII characters “i t n e”.

Bits	Description
31:0	**Vendor**. The 12 8-bit ASCII character codes to create the string “AuthenticAMD”.

CPUID Fn0000_0001_EAX Family, Model, Stepping Identifiers

Also see CPUID Fn8000_0001_EAX [Family, Model, Stepping Identifiers].

Family is an 8-bit value and is defined as: **Family[7:0] =** ({0000b,BaseFamily[3:0]} + ExtendedFamily[7:0]) (e.g., If BaseFamily[3:0] = Fh and ExtendedFamily[7:0] = 07h, then Family[7:0]=16h).

Model is an 8-bit value and is defined as: **Model[7:0]** = {ExtendedModel[3:0], BaseModel[3:0]} (e.g., If ExtendedModel[3:0] = Eh and BaseModel[3:0] = 8h, then Model[7:0] = E8h. Model numbers vary with product).

Bits	Description
31:28	Reserved.
27:20	**ExtFamily: extended family**. Alias of D18F3xFC[ExtFamily].
19:16	**ExtModel: extended model**. Alias of D18F3xFC[ExtModel].
15:12	Reserved.
11:8	**BaseFamily**. Alias of D18F3xFC[BaseFamily].
7:4	**BaseModel**. Alias of D18F3xFC[BaseModel].
3:0	**Stepping**. Alias of D18F3xFC[Stepping].

CPUID Fn0000_0001_EBX LocalApicId, LogicalProcessorCount, CLFlush

Bits	Description
31:24	**LocalApicId: initial local APIC physical ID**. The initial APIC20[ApicId] value. See 2.4.4 [Processor Cores and Downcoring].
23:16	**LogicalProcessorCount: logical processor count**. Value: CPUID Fn8000_0008_ECX[NC] + 1. Specifies the number of cores in the processor as CPUID Fn8000_0008_ECX[NC] + 1.
15:8	**CLFlush: CLFLUSH size in quadwords**. Value: 08h.
7:0	Reserved.

CPUID Fn0000_0001_ECX Feature Identifiers

These values can be over-written by MSRC001_1004.

Bits	Description
31	RAZ. Reserved for use by hypervisor to indicate guest status.
30	**RDRAND: RDRAND instruction support**. Value: 1.
29	**F16C: half-precision convert instruction support**. Value: 1.
28	**AVX: AVX instruction support**. Value: 1. Value: 1.
27	**OSXSAVE: OS enabled support for XGETBV/XSETBV**. 1=The OS has enabled support for XGETBV/XSETBV instructions to query processor extended states.
26	**XSAVE: XSAVE (and related) instruction support**. Value: 1. 1=Support provided for the XSAVE, XRSTOR, XSETBV, and XGETBV instructions and the XFEATURE_ENABLED_MASK register.
25	**AES: AES instruction support**. Value: Product-specific.
24	Reserved.
23	**POPCNT: POPCNT instruction**. Value: 1.
22	**MOVBE: MOVBE instruction support**. Value: 1.
21	**x2APIC: x2APIC capability**. Value: 0.
20	**SSE42: SSE4.2 instruction support**. Value: 1.
19	**SSE41: SSE4.1 instruction support**. Value: 1.
18	Reserved.
17	**PCID: process context identifiers support**. Value: 0.
16:14	Reserved.
13	**CMPXCHG16B: CMPXCHG16B instruction**. Value: 1.
12	**FMA: FMA instruction support**. Value: 1.
11:10	Reserved.
9	**SSSE3: supplemental SSE3 extensions**. Value: 1.
8:4	Reserved.
3	**Monitor: Monitor/Mwait instructions**. Value: ~MSRC001_0015[MonMwaitDis].
2	Reserved.
1	**PCLMULQDQ**: **PCLMULQDQ instruction support**. Value: Product-specific.
0	**SSE3: SSE3 extensions**. Value: 1.

CPUID Fn0000_0001_EDX Feature Identifiers

These values can be over-written by MSRC001_1004.

Bits	Description
31:29	Reserved.

28	**HTT: hyper-threading technology**. Value: CPUID Fn8000_0008_ECX[NC] != 0. 1=Multi-core product (CPUID Fn8000_0008_ECX[NC] != 0). 0=Single core product (CPUID Fn8000_0008_ECX[NC] == 0).
27	Reserved.
26	**SSE2: SSE2 extensions**. Value: 1.
25	**SSE: SSE extensions**. Value: 1.
24	**FXSR: FXSAVE and FXRSTOR instructions**. Value: 1.
23	**MMX™: MMX instructions**. Value: 1.
22:20	Reserved.
19	**CLFSH: CLFLUSH instruction**. Value: 1.
18	Reserved.
17	**PSE36: page-size extensions**. Value: 1.
16	**PAT: page attribute table**. Value: 1.
15	**CMOV: conditional move instructions, CMOV, FCOMI, FCMOV**. Value: 1.
14	**MCA: machine check architecture, MCG_CAP**. Value: 1.
13	**PGE: page global extension, CR4.PGE**. Value: 1.
12	**MTRR: memory-type range registers**. Value: 1.
11	**SysEnterSysExit: SYSENTER and SYSEXIT instructions**. Value: 1.
10	Reserved.
9	**APIC: advanced programmable interrupt controller (APIC) exists and is enabled**. Value: MSR0000_001B[ApicEn].
8	**CMPXCHG8B: CMPXCHG8B instruction**. Value: 1.
7	**MCE: machine check exception, CR4.MCE**. Value: 1.
6	**PAE: physical-address extensions (PAE)**. Value: 1.
5	**MSR: AMD model-specific registers (MSRs), with RDMSR and WRMSR instructions**. Value: 1.
4	**TSC: time stamp counter, RDTSC/RDTSCP instructions, CR4.TSD**. Value: 1.
3	**PSE: page-size extensions (4 MB pages)**. Value: 1.
2	**DE: debugging extensions, IO breakpoints, CR4.DE**. Value: 1.
1	**VME: virtual-mode enhancements**. Value: 1.
0	**FPU: x87 floating point unit on-chip**. Value: 1.

CPUID Fn0000_000[4:2] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_0005_EAX Monitor/MWait

Bits	Description
31:16	Reserved.
15:0	**MonLineSizeMin: smallest monitor-line size in bytes**. Value: 40h.

CPUID Fn0000_0005_EBX Monitor/MWait

Bits	Description
31:16	Reserved.
15:0	**MonLineSizeMax: largest monitor-line size in bytes**. Value: 40h.

CPUID Fn0000_0005_ECX Monitor/MWait

Bits	Description
31:2	Reserved.
1	**IBE: interrupt break-event**. Value: 1.
0	**EMX: enumerate MONITOR/MWAIT extensions**. Value: 1.

CPUID Fn0000_0005_EDX Monitor/MWait

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_EAX Thermal and Power Management

Bits	Description
31:3	Reserved.
2	**ARAT: always running APIC timer**. Value: 1. 1=Indicates support for APIC timer always running feature.
1:0	Reserved.

CPUID Fn0000_0006_EBX Thermal and Power Management

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_ECX Thermal and Power Management

These values can be over-written by MSRC001_1003.

Bits	Description
31:1	Reserved.
0	**EffFreq: effective frequency interface**. Value: 1. 1=Indicates presence of MSR0000_00E7 [Max Performance Frequency Clock Count (MPERF)] and MSR0000_00E8 [Actual Performance Frequency Clock Count (APERF)].

CPUID Fn0000_0006_EDX Thermal and Power Management

Bits	Description
31:0	Reserved.

CPUID Fn0000_0007_EAX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:0	Reserved.

CPUID Fn0000_0007_EBX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:11	Reserved.
10	**INVPCID: invalidate processor context ID**. Value: 0.
9	Reserved.
8	**BMI2: bit manipulation group 2 instruction support**. Value: 1.
7	**SMEP: supervisor mode execution protection**. Value: 1.
6	Reserved.
5	**AVX2: AVX extension support**. Value: 1.
4	Reserved.
3	**BMI1: bit manipulation group 1 instruction support**. Value: 1.
2:1	Reserved.
0	**FSGSBASE: FS and GS base read write instruction support**. Value: 1.

CPUID Fn0000_0007_ECX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:0	Reserved.

CPUID Fn0000_0007_EDX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:0	Reserved.

CPUID Fn0000_000[A:8] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_000B Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_000C Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EAX_x0 Processor Extended State Enumeration (ECX=0)

Bits	Description
31:0	**XFeatureSupportedMask[31:0]**. Value: 0000_0007h.

CPUID Fn0000_000D_EBX_x0 Processor Extended State Enumeration (ECX=0)

Bits	Description
31:0	**XFeatureEnabledSizeMax**. Size in bytes of XSAVE/XRSTOR area for the currently enabled features in XCR0. Value: 512 + 64 + (IF (XCR0[AVX] \| XCR0[LWP]) THEN 256. ELSE 0. ENDIF.) + (IF XCR0[LWP] THEN 128. ELSE 0. ENDIF.) The components of this sum are described as follows: • 512: FPU/SSE save area (needed even if XCR0[SSE] == 0). • 64: Header size (always needed). • Size of YMM area if YMM enabled. • Size of LWP area if LWP enabled.

CPUID Fn0000_000D_ECX_x0 Processor Extended State Enumeration (ECX=0)

Bits	Description
31:0	**XFeatureSupportedSizeMax**. Size in bytes of XSAVE/XRSTOR area for all features that the core supports. See XFeatureEnabledSizeMax. Value: 0000_03C0h.

CPUID Fn0000_000D_EDX_x0 Processor Extended State Enumeration (ECX=0)

Bits	Description
31:0	**XFeatureSupportedMask[63:32]**. Value: 4000_0000h.

CPUID Fn0000_000D_EAX_x1 Processor Extended State Enumeration (ECX=1)

Bits	Description
31:1	Reserved.
0	**XSAVEOPT: XSAVEOPT is available**. Value: 1.

CPUID Fn0000_000D_E[D,C,B]X_x1 Processor Extended State Enumeration (ECX=1)

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EAX_x2 Processor Extended State Enumeration (ECX=2)

Bits	Description
31:0	**YmmSaveStateSize: YMM save state byte size**. Value: 0000_0100h.

CPUID Fn0000_000D_EBX_x2 Processor Extended State Enumeration (ECX=2)

Bits	Description
31:0	**YmmSaveStateOffset: YMM save state byte offset**. Value: 0000_0240h.

CPUID Fn0000_000D_ECX_x2 Processor Extended State Enumeration (ECX=2)

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EDX_x2 Processor Extended State Enumeration (ECX=2)

Bits	Description
31:0	Reserved.

For CPUID Fn0000_000D, if ECX>2 and ECX<62 then EAX/EBX/ECX/EDX will return 0.

CPUID Fn0000_000D_EAX_x3E Processor Extended State Enumeration (ECX=62)

Bits	Description
31:0	**LwpSaveStateSize: LWP save state byte size**. Value: 0000_0080h.

CPUID Fn0000_000D_EBX_x3E Processor Extended State Enumeration (ECX=62)

Bits	Description
31:0	**LwpSaveStateOffset: LWP save state byte offset**. Value: 0000_0340h.

CPUID Fn0000_000D_ECX_x3E Processor Extended State Enumeration (ECX=62)

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EDX_x3E Processor Extended State Enumeration (ECX=62)

Bits	Description
31:0	Reserved.

For CPUID Fn0000_000D, if ECX>62 then EAX/EBX/ECX/EDX returns 0.

CPUID Fn8000_0000_EAX Largest Extended Function Number

Bits	Description
31:0	**LFuncExt: largest extended function**. Value: 8000_001Eh. The largest CPUID extended function input value supported by the processor implementation.

CPUID Fn8000_0000_E[D,C,B]X Processor Vendor

CPUID Fn0000_0000_E[D,C,B]X and CPUID Fn8000_0000_E[D,C,B]X return the same value.

Table 231: CPUID Fn8000_0000_E[B,C,D]X Value

Register	Value	Description
CPUID Fn8000_0000_EBX	6874_7541h	The ASCII characters “h t u A”.
CPUID Fn8000_0000_ECX	444D_4163h	The ASCII characters “D M A c”.
CPUID Fn8000_0000_EDX	6974_6E65h	The ASCII characters “i t n e”.

Bits	Description
31:0	**Vendor**. The 12 8-bit ASCII character codes to create the string “AuthenticAMD”.

CPUID Fn8000_0001_EAX Family, Model, Stepping Identifiers

Also see CPUID Fn0000_0001_EAX [Family, Model, Stepping Identifiers].

Bits	Description
31:28	Reserved.
27:20	**ExtFamily: extended family**. CPUID Fn8000_0001_EAX[ExtFamily] is an alias of D18F3xFC[ExtFamily].
19:16	**ExtModel: extended model**. CPUID Fn8000_0001_EAX[ExtModel] is an alias of D18F3xFC[ExtModel].
15:12	Reserved.
11:8	**BaseFamily**. CPUID Fn8000_0001_EAX[BaseFamily] is an alias of D18F3xFC[BaseFamily].
7:4	**BaseModel**. CPUID Fn8000_0001_EAX[BaseModel] is an alias of D18F3xFC[BaseModel].
3:0	**Stepping**. CPUID Fn8000_0001_EAX[Stepping] is an alias of D18F3xFC[Stepping].

CPUID Fn8000_0001_EBX BrandId Identifier

Bits	Description
31:28	**PkgType: package type**. Specifies the package type. Value: Product-specific. Bits — Description 0h — FP4 (BGA) 1h — Reserved. 2h — AM4 (uPGA) 3h — Reserved 4h — FT4 (BGA) Fh-5h — Reserved.
27:0	Reserved.

CPUID Fn8000_0001_ECX Feature Identifiers

These values can be over-written by MSRC001_1005.

Bits	Description
31:30	Reserved.
29	**MwaitExtended: MWAITX/MONITORX support**. Value: ~(MSRC001_0015[MonMwaitDis]). 1=MWAITX and MONITORX capability is supported.
28	**PerfCtrExtL2I: L2I performance counter extensions support**. Value: 0. Indicates support for MSRC001_023[6,4,2,0] and MSRC001_023[7,5,3,1].
27	**PerfTsc: performance time-stamp counter supported**. Value: 1. Indicates support for MSRC001_0280 [Performance Time Stamp Counter (CU_PTSC)].
26	**DataBreakpointExtension**. Value: 1. Indicates data breakpoint support for MSRC001_1027 and MSRC001_101[B:9].
25	Reserved.
24	**PerfCtrExtNB: NB performance counter extensions support.** Value: 1. Indicates support for MSRC001_024[6,4,2,0] and MSRC001_024[7,5,3,1].
23	**PerfCtrExtCore: core performance counter extensions support**. Value: 1. Indicates support for MSRC001_020[A,8,6,4,2,0] and MSRC001_020[B,9,7,5,3,1].
22	**TopologyExtensions: topology extensions support**. Value: 1. Indicates support for CPUID Fn8000_001D_EAX_x0-CPUID Fn8000_001E_EDX.
21	**TBM: trailing bit manipulation instruction support**. Value: 1.
20:18	Reserved.
17	**TCE: translation cache extension**. Value: 1.
16	**FMA4: 4-operand FMA instruction support**. Value: 1.
15	**LWP: lightweight profiling support**. Value: 1.
14	Reserved.
13	**WDT: watchdog timer support**. Value: 1.
12	**SKINIT: SKINIT and STGI support**. Value: 1.
11	**XOP: extended operation support**. Value: 1.
10	**IBS: Instruction Based Sampling**. Value: 1.
9	**OSVW: OS Visible Work-around support**. Value: 1.
8	**3DNowPrefetch: Prefetch and PrefetchW instructions**. Value: 1.
7	**MisAlignSse: Misaligned SSE Mode.** Value: 1.
6	**SSE4A: EXTRQ, INSERTQ, MOVNTSS, and MOVNTSD instruction support.** Value: 1.
5	**ABM: advanced bit manipulation**. Value: 1. LZCNT instruction support.
4	**AltMovCr8: LOCK MOV CR0 means MOV CR8**. Value: 1.
3	**ExtApicSpace: extended APIC register space**. Value: 1.
2	**SVM: Secure Virtual Mode feature**. Value: Product-specific. Indicates support for: VMRUN, VMLOAD, VMSAVE, CLGI, VMMCALL, and INVLPGA.

1	**CmpLegacy: core multi-processing legacy mode**. Value: Product-specific. 1=Multi-core product (CPUID Fn8000_0008_ECX[NC] != 0). 0=Single core product (CPUID Fn8000_0008_ECX[NC] == 0).
0	**LahfSahf: LAHF/SAHF instructions**. Value: 1.

CPUID Fn8000_0001_EDX Feature Identifiers

These values can be over-written by MSRC001_1005.

Bits	Description
31	**3DNow: 3DNow!™ instructions**. Value: 0.
30	**3DNowExt: AMD extensions to 3DNow!™ instructions**. Value: 0.
29	**LM: long mode**. Value: 1.
28	Reserved.
27	**RDTSCP: RDTSCP instruction**. Value: 1.
26	**Page1GB: one GB large page support**. Value: 1.
25	**FFXSR: FXSAVE and FXRSTOR instruction optimizations**. Value: 1.
24	**FXSR: FXSAVE and FXRSTOR instructions**. Value: 1.
23	**MMX™: MMX instructions**. Value: 1.
22	**MmxExt: AMD extensions to MMX instructions**. Value: 1.
21	Reserved.
20	**NX: no-execute page protection**. Value: 1.
19:18	Reserved.
17	**PSE36: page-size extensions**. Value: 1.
16	**PAT: page attribute table**. Value: 1.
15	**CMOV: conditional move instructions, CMOV, FCOMI, FCMOV**. Value: 1.
14	**MCA: machine check architecture, MCG_CAP**. Value: 1.
13	**PGE: page global extension, CR4.PGE**. Value: 1.
12	**MTRR: memory-type range registers**. Value: 1.
11	**SysCallSysRet: SYSCALL and SYSRET instructions**. Value: 1.
10	Reserved.
9	**APIC: advanced programmable interrupt controller (APIC) exists and is enabled**. Value: MSR0000_001B[ApicEn].
8	**CMPXCHG8B: CMPXCHG8B instruction**. Value: 1.
7	**MCE: machine check exception, CR4.MCE**. Value: 1.
6	**PAE: physical-address extensions (PAE)**. Value: 1.
5	**MSR: model-specific registers (MSRs), with RDMSR and WRMSR instructions**. Value: 1.
4	**TSC: time stamp counter, RDTSC/RDTSCP instructions, CR4.TSD**. Value:1.
3	**PSE: page-size extensions (4 MB pages)**. Value: 1.
2	**DE: debugging extensions, IO breakpoints, CR4.DE**. Value: 1.

1	**VME: virtual-mode enhancements**. Value: 1.
0	**FPU: x87 floating point unit on-chip**. Value: 1.

CPUID Fn8000_000[4:2]_E[D,C,B,A]X Processor Name String Identifier

Table 232: Valid Values for CPUID Fn8000_000[4:2]_E[D,C,B,A]X

Register	Value
CPUID Fn8000_0002_EAX	MSRC001_0030[31:0]
CPUID Fn8000_0002_EBX	MSRC001_0030[63:32]
CPUID Fn8000_0002_ECX	MSRC001_0031[31:0]
CPUID Fn8000_0002_EDX	MSRC001_0031[63:32]
CPUID Fn8000_0003_EAX	MSRC001_0032[31:0]
CPUID Fn8000_0003_EBX	MSRC001_0032[63:32]
CPUID Fn8000_0003_ECX	MSRC001_0033[31:0]
CPUID Fn8000_0003_EDX	MSRC001_0033[63:32]
CPUID Fn8000_0004_EAX	MSRC001_0034[31:0]
CPUID Fn8000_0004_EBX	MSRC001_0034[63:32]
CPUID Fn8000_0004_ECX	MSRC001_0035[31:0]
CPUID Fn8000_0004_EDX	MSRC001_0035[63:32]

Bits	Description
31:0	**ProcName: processor name**. These return the ASCII string corresponding to the processor name, stored in MSRC001_00[35:30] [Processor Name String].

CPUID Fn8000_0005_EAX L1 TLB 2M/4M Identifiers

This function provides the processor's first level cache and TLB characteristics for each core.

Bits	Description
31:24	**L1DTlb2and4MAssoc: data TLB associativity for 2 MB and 4 MB pages**. Value: FFh. See: CPUID Fn8000_0005_ECX[L1DcAssoc].
23:16	**L1DTlb2and4MSize: data TLB number of entries for 2 MB and 4 MB pages**. Value: 64. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.
15:8	**L1ITlb2and4MAssoc: instruction TLB associativity for 2 MB and 4 MB pages**. Value: FFh. See: CPUID Fn8000_0005_ECX[L1DcAssoc].
7:0	**L1ITlb2and4MSize: instruction TLB number of entries for 2 MB and 4 MB pages**. Value: 24. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.

CPUID Fn8000_0005_EBX L1 TLB 4K Identifiers

See: CPUID Fn8000_0005_EAX.

Bits	Description
31:24	**L1DTlb4KAssoc: data TLB associativity for 4 KB pages**. Value: FFh. See: CPUID Fn8000_0005_ECX[L1DcAssoc].
23:16	**L1DTlb4KSize: data TLB number of entries for 4 KB pages**. Value: 64.
15:8	**L1ITlb4KAssoc: instruction TLB associativity for 4 KB pages**. Value: FFh. ITLB associativity for 4 KB pages is reported by CPUID Fn8000_0006_EBX[L2ITlb4KAssoc].
7:0	**L1ITlb4KSize: instruction TLB number of entries for 4 KB pages**. Value: 48. ITLB size for 4 KB pages is reported by CPUID Fn8000_0006_EBX[L2ITlb4KSize].

CPUID Fn8000_0005_ECX L1 Data Cache Identifiers

This function provides first level cache characteristics for each core.

Bits	Description
31:24	**L1DcSize: L1 data cache size in KB**. Value: 32.
23:16	**L1DcAssoc: L1 data cache associativity**. Value: 8. Bits — Description 00h — Reserved 01h — 1 way (direct mapped) 02h — 2 way 03h — 3 way FEh-04h — [L1IcAssoc] way FFh — Fully associative
15:8	**L1DcLinesPerTag: L1 data cache lines per tag**. Value: 1.
7:0	**L1DcLineSize: L1 data cache line size in bytes**. Value: 64.

CPUID Fn8000_0005_EDX L1 Instruction Cache Identifiers

This function provides first level cache characteristics for each core.

Bits	Description
31:24	**L1IcSize: L1 instruction cache size KB**. Value: 96.
23:16	**L1IcAssoc: L1 instruction cache associativity**. Value: 3. See: CPUID Fn8000_0005_ECX[L1DcAssoc].
15:8	**L1IcLinesPerTag: L1 instruction cache lines per tag**. Value: 1.
7:0	**L1IcLineSize: L1 instruction cache line size in bytes**. Value: 64.

CPUID Fn8000_0006_EAX L2 TLB 2M/4M Identifiers

This function provides the processor's second level cache and TLB characteristics for each core.

Bits	Description
31:28	**L2DTlb2and4MAssoc: L2 data TLB associativity for 2 MB and 4 MB pages**. Value: 6. See: CPUID Fn8000_0006_ECX[L2Assoc].
27:16	**L2DTlb2and4MSize: L2 data TLB number of entries for 2 MB and 4 MB pages**. Value: 1024. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.
15:12	**L2ITlb2and4MAssoc: L2 instruction TLB associativity for 2 MB and 4 MB pages**. Value: 6. See: CPUID Fn8000_0006_ECX[L2Assoc].
11:0	**L2ITlb2and4MSize: L2 instruction TLB number of entries for 2 MB and 4 MB pages**. Value: 1024. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.

CPUID Fn8000_0006_EBX L2 TLB 4K Identifiers

This function provides second level TLB characteristics for 4K pages shared by each core on a Compute Unit.

Bits	Description
31:28	**L2DTlb4KAssoc: L2 data TLB associativity for 4 KB pages**. **Value: 6.** See: CPUID Fn8000_0006_ECX[L2Assoc].
27:16	**L2DTlb4KSize: L2 data TLB number of entries for 4 KB pages**. Value: 1024.
15:12	**L2ITlb4KAssoc: L2 instruction TLB associativity for 4 KB pages**. Value: 4. See: CPUID Fn8000_0006_ECX[L2Assoc].
11:0	**L2ITlb4KSize: L2 instruction TLB number of entries for 4 KB pages**. Value: 512.

CPUID Fn8000_0006_ECX L2 Cache Identifiers

Bits	Description
31:16	**L2Size: L2 cache size in KB**. Value: Product-specific. Bits — Description 03FFh-0000h — Reserved. 0400h — 1 MB 07FFh-0401h — Reserved. 0800h — 2 MB FFFFh-0801h — Reserved.

15:12	**L2Assoc: L2 cache associativity**. Value: 8. Bits / Description / Bits / Description 0h Disabled. 8h 16 ways 1h 1 way (direct mapped) 9h Reserved 2h 2 ways Ah 32 ways 3h Reserved. Bh 48 ways 4h 4 ways Ch 64 ways 5h Reserved. Dh 96 ways 6h 8 ways Eh 128 ways 7h Reserved. Fh Fully associative
11:8	**L2LinesPerTag: L2 cache lines per tag**. Value: 1.
7:0	**L2LineSize: L2 cache line size in bytes**. Value: 64.

CPUID Fn8000_0006_EDX L3 Cache Identifiers

This function provides third level cache characteristics shared by all cores.

Bits	Description
31:18	**L3Size: L3 cache size**. Value: 0.
17:16	Reserved.
15:12	**L3Assoc: L3 cache associativity**. Value: 0.
11:8	**L3LinesPerTag: L3 cache lines per tag**. Value: 0.
7:0	**L3LineSize: L3 cache line size in bytes**. Value: 64.

CPUID Fn8000_0007_EAX Processor Feedback Capabilities

Bits	Description
31:16	**MaxWrapTime**. Value: 0000h. Specifies the maximum time between reads that software should use to avoid two wraps. A read of at least once every MaxWrapTime seconds results in either zero or one wrap during that interval.
15:8	**Version**. Value: 00h. Specifies the processor feedback capabilities version.
7:0	**NumberOfMonitors**. Value: 00h. Specifies the number of processor feedback MSR pairs supported. Valid if (CPUID Fn8000_0007_EDX[ProcFeedbackInterface] == 1).

CPUID Fn8000_0007_EBX RAS Capabilities

Bits	Description
31:3	Reserved.
2	**HWA: hardware assert supported**. 1=Indicates support for Hardware Assert Machine Check Architecture. Value: 1.

1	**SUCCOR: Software uncorrectable error containment and recovery capability**. Value: 0.
0	**McaOverflowRecov: MCA overflow recovery support**. Value: 1. 1=MCA overflow conditions (MCi_STATUS[Overflow]=1) are not fatal; software may safely ignore such conditions. 0=MCA overflow conditions require software to shut down the system. See 2.16.1.6 [Handling Machine Check Exceptions].

CPUID Fn8000_0007_ECX Advanced Power Management Information

Bits	Description
31:0	**CmpUnitPwrSampleTimeRatio**. Specifies the compute unit power accumulator sample period.

CPUID Fn8000_0007_EDX Advanced Power Management Information

This function provides advanced power management feature identifiers.

Bits	Description
31:12	Reserved.
11	**ProcFeedbackInterface: processor feedback interface**. Value: 0. 1=Indicates support for processor feedback interface; CPUID Fn8000_0007_EAX.
10	**EffFreqRO: read-only effective frequency interface**. Value: 1. 1=Indicates presence of MSRC000_00E7 [Read-Only Max Performance Frequency Clock Count (MPerfReadOnly)] and MSRC000_00E8 [Read-Only Actual Performance Frequency Clock Count (APerfReadOnly)].
9	**CPB: core performance boost**. Value: Product-specific. 1=Indicates presence of MSRC001_0015[CpbDis] and support for core performance boost. See 2.5.8 [Application Power Management (APM)].
8	**TscInvariant: TSC invariant**. Value: 1. The TSC rate is invariant.
7	**HwPstate: hardware P-state control**. Value: 1. MSRC001_0061 [P-state Current Limit], MSRC001_0062 [P-state Control] and MSRC001_0063 [P-state Status] exist.
6	**100MHzSteps: 100 MHz multiplier Control**. Value: 1.
5	Reserved.
4	**TM: hardware thermal control (HTC)**. Value: Product-specific.
3	**TTP: THERMTRIP**. Value: 1.
2	**VID: Voltage ID control**. Value: 0. Function replaced by HwPstate.
1	**FID: Frequency ID control**. Value: 0. Function replaced by HwPstate.
0	**TS: Temperature sensor**. Value: 1.

CPUID Fn8000_0008_EAX Long Mode Address Size Identifiers

This provides information about the maximum physical and linear address width supported by the processor.

Bits	Description
31:24	Reserved.

23:16	**GuestPhysAddrSize: maximum guest physical byte address size in bits**. Value: 0. 0=The maximum guest physical address size defined by PhysAddrSize.
15:8	**LinAddrSize: Maximum linear byte address size in bits**. Value: IF (CPUID Fn8000_0001_EDX[LM]) THEN 30h. ELSE 20h. ENDIF.
7:0	**PhysAddrSize: Maximum physical byte address size in bits**. Value: 30h.

CPUID Fn8000_0008_EBX Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_0008_ECX Size Identifiers

This provides information about the number of cores supported by the processor.

Bits	Description
31:18	Reserved.
17:16	**PerfTscSize: performance time-stamp counter size**. Value: 00b. Indicates the size of MSRC001_0280[PTSC]. Valid only when (CPUID Fn8000_0001_ECX[PerfTsc] == 1). Bits — Description 00b — 40 bits 01b — 48 bits 10b — 56 bits 11b — 64 bits
15:12	**ApicIdCoreIdSize: APIC ID size**. Value: 4h. The number of bits in the initial APIC20[ApicId] value that indicate core ID within a processor.
11:8	Reserved.
7:0	**NC: number of cores - 1**. Value: D18F5x84[CmpCap]-COUNT(D18F3x190[DisCore]). The number of cores in the processor is NC+1 (e.g., if NC=0, then there is one core). See 2.4.4 [Processor Cores and Downcoring].

CPUID Fn8000_0008_EDX Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_0009 Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_000A_EAX SVM Revision and Feature Identification

This provides SVM revision. If (CPUID Fn8000_0001_ECX[SVM] == 0) then CPUID Fn8000_000A_EAX is reserved.

Bits	Description
31:8	Reserved.
7:0	**SvmRev: SVM revision**. Value: 01h.

CPUID Fn8000_000A_EBX SVM Revision and Feature Identification

This provides SVM revision and feature information. If (CPUID Fn8000_0001_ECX[SVM] == 0) then CPUID Fn8000_000A_EBX is reserved.

Bits	Description
31:0	**NASID: number of address space identifiers (ASID)**. Value: 8000h.

CPUID Fn8000_000A_ECX SVM Revision and Feature Identification

Bits	Description
31:0	Reserved.

CPUID Fn8000_000A_EDX SVM Revision and Feature Identification

This provides SVM feature information. If (CPUID Fn8000_0001_ECX[SVM] == 0) then CPUID Fn8000_000A_EDX is reserved.

Bits	Description
31:14	Reserved.
13	**AVIC: AMD virtual interrupt controller**. Value: 1. 1=Support indicated for SVM mode virtualized interrupt controller; Indicates support for MSRC001_011B [AVIC Doorbell].
12	**PauseFilterThreshold: PAUSE filter threshold**. Value: 1.
11	Reserved.
10	**PauseFilter: pause intercept filter**. Value: 1.
9:8	Reserved.
7	**DecodeAssists: decode assists**. Value: 1.
6	**FlushByAsid: flush by ASID**. Value: 1.
5	**VmcbClean: VMCB clean bits**. Value: 1.
4	**TscRateMsr: MSR based TSC rate control**. Value: 1. 1=Indicates support for TSC ratio MSRC000_0104.
3	**NRIPS: NRIP Save**. Value: 1.
2	**SVML: SVM lock**. Value: 1.

1	**LbrVirt: LBR virtualization**. Value: 1.
0	**NP: nested paging**. Value: 1.

CPUID Fn8000_00[18:0B] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_0019_EAX L1 TLB 1G Identifiers

This function provides first level TLB characteristics for 1GB pages shared by each core on a Compute Unit.

Bits	Description
31:28	**L1DTlb1GAssoc: L1 data TLB associativity for 1 GB pages**. See: CPUID Fn8000_0006_ECX[L2Assoc]. Value: Fh.
27:16	**L1DTlb1GSize: L1 data TLB number of entries for 1 GB pages**. Value: 64.
15:12	**L1ITlb1GAssoc: L1 instruction TLB associativity for 1 GB pages**. See: CPUID Fn8000_0006_ECX[L2Assoc]. Value: Fh.
11:0	**L1ITlb1GSize: L1 instruction TLB number of entries for 1 GB pages**. Value: 24.

CPUID Fn8000_0019_EBX L2 TLB 1G Identifiers

This provides 1 GB paging information. The *associativity* fields are defined by CPUID Fn8000_0006_EAX, CPUID Fn8000_0006_EBX, CPUID Fn8000_0006_ECX and CPUID Fn8000_0006_EDX.

Bits	Description
31:28	**L2DTlb1GAssoc: L2 data TLB associativity for 1 GB pages**. See: CPUID Fn8000_0006_ECX[L2Assoc]. Value: 6.
27:16	**L2DTlb1GSize: L2 data TLB number of entries for 1 GB pages**. Value: 1024.
15:12	**L2ITlb1GAssoc: L2 instruction TLB associativity for 1 GB pages**. See: CPUID Fn8000_0006_ECX[L2Assoc]. Value: 6.
11:0	**L2ITlb1GSize: L2 instruction TLB number of entries for 1 GB pages**. Value: 1024.

CPUID Fn8000_0019_E[D,C]X Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_001A_EAX Performance Optimization Identifiers

This function returns performance related information. For more details on how to use these bits to optimize software, see the optimization guide.

Bits	Description
31:3	Reserved.

2	**FP256**. Value: 0.
1	**MOVU**. Value: 1.
0	**FP128**. Value: 1.

CPUID Fn8000_001A_E[D,C,B]X Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_001B_EAX Instruction Based Sampling Identifiers

This function returns IBS feature information.

Bits	Description
31:11	Reserved.
10	**IbsOpData4: IBS op data 4 MSR supported**. Value: 1. See MSRC001_103D [IBS Op Data 4 (DC_IBS_DATA2)].
9	**IbsFetchCtlExtd: IBS fetch control extended MSR supported**. Value: 1. 1=Indicates support for MSRC001_103C [IBS Fetch Control Extended (IC_IBS_EXTD_CTL)].
8	**OpBrnFuse: fused branch micro-op indication supported**. Value: 1. 1=Indicates support for MSRC001_1035[IbsOpBrnFuse].
7	**RipInvalidChk: invalid RIP indication supported**. Value: 1. 1=Indicates support for MSRC001_1035[IbsRipInvalid].
6	**OpCntExt: IbsOpCurCnt and IbsOpMaxCnt extend by 7 bits**. Value: 1. 1=Indicates support for MSRC001_1033[IbsOpCurCnt[26:20], IbsOpMaxCnt[26:20]].
5	**BrnTrgt: branch target address reporting supported**. Value: 1.
4	**OpCnt: op counting mode supported**. Value: 1.
3	**RdWrOpCnt: read write of op counter supported**. Value: 1.
2	**OpSam: IBS execution sampling supported**. Value: 1.
1	**FetchSam: IBS fetch sampling supported**. Value: 1.
0	**IBSFFV: IBS feature flags valid**. Value: 1.

CPUID Fn8000_001B_E[D,C,B]X Instruction Based Sampling Identifiers

Bits	Description
31:0	Reserved.

CPUID Fn8000_001C_EAX Lightweight Profiling Capabilities 0

This function returns IBS feature information; see the Lightweight Profiling Specification section titled "Detecting LWP". If (CPUID Fn8000_0001_ECX[LWP] == 0) then CPUID Fn8000_001C_E[D,C,B,A]X is reserved.

Bits	Description
31	**LwpInt: interrupt on threshold overflow available**. Value: MSRC000_0105[LwpInt]. 1=Interrupt on threshold overflow is available.
30	**LwpPTSC: performance time stamp counter in event record is available**. Value: MSRC000_0105[LwpPTSC]. 1=Performance time stamp counter in event record is available.
29	**LwpCont: sampling in continuous mode is available**. Value: MSRC000_0105[LwpCont]. 1=Sampling in continuous mode is available.
28:7	Reserved.
6	**LwpRNH: core reference clocks not halted event available**. Value: MSRC000_0105[LwpRNH]. 1=Core reference clocks not halted event is available.
5	**LwpCNH: core clocks not halted event available**. Value: MSRC000_0105[LwpCNH]. 1=Core clocks not halted event is available.
4	**LwpDME: DC miss event available**. Value: MSRC000_0105[LwpDME]. 1=DC miss event is available.
3	**LwpBRE: branch retired event available**. Value: MSRC000_0105[LwpBRE]. 1=Branch retired event is available.
2	**LwpIRE: instructions retired event available**. Value: MSRC000_0105[LwpIRE]. 1=Instructions retired event is available.
1	**LwpVAL: LWPVAL instruction available**. Value: MSRC000_0105[LwpVAL]. 1=LWPVAL instruction is available.
0	**LwpAvail: LWP available**. Value: XCR0[62]. 1=LWP is available.

CPUID Fn8000_001C_EBX Lightweight Profiling Capabilities 0

See CPUID Fn8000_001C_EAX.

Bits	Description
31:24	**LwpEventOffset: offset to the EventInterval1 field**. Value: 80h. Offset from the start of the LWPCB to the EventInterval1 field.
23:16	**LwpMaxEvents: maximum EventId**. Value: 3. Maximum EventId value that is supported.
15:8	**LwpEventSize: event record size**. Value: 20h. Size in bytes of an event record in the LWP event ring buffer.
7:0	**LwpCbSize: control block size**. Value: 13h. Size in quadwords of the LWPCB.

CPUID Fn8000_001C_ECX Lightweight Profiling Capabilities 0

See CPUID Fn8000_001C_EAX.

Bits	Description
31	**LwpCacheLatency: cache latency filtering supported**. Value: 0. 1=Cache-related events can be filtered by latency.
30	**LwpCacheLevels: cache level filtering supported**. Value: 0. 1=Cache-related events can be filtered by the cache level that returned the data.
29	**LwpIpFiltering: IP filtering supported**. Value: 0. 1=IP filtering is supported.
28	**LwpBranchPrediction: branch prediction filtering supported**. Value: 0. 1=Branches Retired events can be filtered based on whether the branch was predicted properly.
27:24	Reserved.
23:16	**LwpMinBufferSize: event ring buffer size**. Value: 01h. Minimum size of the LWP event ring buffer, in units of 32 event records.
15:9	**LwpVersion: version**. Value: 0000001b. Version of LWP implementation.
8:6	**LwpLatencyRnd: amount cache latency is rounded**. Value: 0. The amount by which cache latency is rounded.
5	**LwpDataAddress: data cache miss address valid**. Value: 0. 1=Address is valid for cache miss event records.
4:0	**LwpLatencyMax: latency counter bit size**. Value: 0. Size in bits of the cache latency counters.

CPUID Fn8000_001C_EDX Lightweight Profiling Capabilities 0

See CPUID Fn8000_001C_EAX.

Bits	Description
31	**LwpInt: interrupt on threshold overflow supported**. Value: 1. 1=Interrupt on threshold overflow is supported.
30	**LwpPTSC: performance time stamp counter in event record is available**. Value: 1. 1=Performance time stamp counter in event record is supported.
29	**LwpCont: sampling in continuous mode is available**. Value: 1. 1=Sampling in continuous mode is supported.
28:7	Reserved.
6	**LwpRNH: core reference clocks not halted event supported**. Value: 0. 1=Core reference clocks not halted event is supported.
5	**LwpCNH: core clocks not halted event supported**. Value: 0. 1=Core clocks not halted event is supported.
4	**LwpDME: DC miss event supported**. Value: 0. 1=DC miss event is supported.
3	**LwpBRE: branch retired event supported**. Value: 1. 1=Branch retired event is supported.
2	**LwpIRE: instructions retired event supported**. Value: 1. 1=Instructions retired event is supported.
1	**LwpVAL: LWPVAL instruction supported**. Value: 1. 1=LWPVAL instruction is supported.
0	**LwpAvail: lightweight profiling supported**. Value: 1. 1=Lightweight profiling is supported.

CPUID Fn8000_001D_EAX_x0 Cache Properties

CPUID Fn8000_001D_EAX_x0 reports topology information for the DC.
If (CPUID Fn8000_0001_ECX[TopologyExtensions] == 0) then CPUID Fn8000_001D_E[D,C,B,A]X is reserved.

Table 233: ECX mapping to Cache Type for CPUID Fn8000_001D_E[D,C,B,A]X

ECX	Cache Type
0	DC
1	IC
2	L2
3	Null

Bits	Description
31:26	Reserved.
25:14	**NumSharingCache: number of cores sharing cache**. Value: 000h. The number of cores sharing this cache is NumSharingCache+1.
13:10	Reserved.
9	**FullyAssociative: fully associative cache**. Value: 0. 1=Cache is fully associative.
8	**SelfInitialization: cache is self-initializing**. Value: 1. 1=Cache is self initializing; cache does not need software initialization.
7:5	**CacheLevel: cache level.** Identifies the cache level. Value: 001b. Bits — Description 000b — Reserved. 001b — Level 1 010b — Level 2 011b — Level 3 111b-100b — Reserved.
4:0	**CacheType: cache type**. Identifies the type of cache. Value: 01h. Bits — Description 00h — Null; no more caches. 01h — Data cache 02h — Instruction cache 03h — Unified cache 1Fh-04h — Reserved.

CPUID Fn8000_001D_EAX_x1 Cache Properties

CPUID Fn8000_001D_EAX_x1 reports topology information for the IC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:26	Reserved.
25:14	**NumSharingCache: number of cores sharing cache**. See: CPUID Fn8000_001D_EAX_x0[NumSharingCache]. Value: IF (DualCoreEnabled == 1) THEN 001h. ELSE 000h. ENDIF.

13:10	Reserved.
9	**FullyAssociative: fully associative cache**. Value: 0. See: CPUID Fn8000_001D_EAX_x0[FullyAssociative].
8	**SelfInitialization: cache is self-initializing**. Value: 1. See: CPUID Fn8000_001D_EAX_x0[SelfInitialization].
7:5	**CacheLevel: cache level**. Identifies the cache level. Value: 001b. See: CPUID Fn8000_001D_EAX_x0[CacheLevel].
4:0	**CacheType: cache type**. Value: 02h. See: CPUID Fn8000_001D_EAX_x0[CacheType].

CPUID Fn8000_001D_EAX_x2 Cache Properties

CPUID Fn8000_001D_EAX_x2 reports topology information for the L2. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:26	Reserved.
25:14	**NumSharingCache: number of cores sharing cache**. See: CPUID Fn8000_001D_EAX_x0[NumSharingCache]. Value: IF (DualCoreEnabled == 1) THEN 001h. ELSE 000h. ENDIF.
13:10	Reserved.
9	**FullyAssociative: fully associative cache**. Value: 0. See: CPUID Fn8000_001D_EAX_x0[FullyAssociative].
8	**SelfInitialization: cache is self-initializing**. Value: 1. See: CPUID Fn8000_001D_EAX_x0[SelfInitialization].
7:5	**CacheLevel: cache level.** Identifies the cache level. Value: 010b. See: CPUID Fn8000_001D_EAX_x0[CacheLevel].
4:0	**CacheType: cache type**. Value: 03h. See: CPUID Fn8000_001D_EAX_x0[CacheType].

CPUID Fn8000_001D_EAX_x3 Cache Properties

CPUID Fn8000_001D_EAX_x3 reports done/null. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:5	Reserved.
4:0	**CacheType: cache type**. Value: 00h. See: CPUID Fn8000_001D_EAX_x0[CacheType].

CPUID Fn8000_001D_EBX_x0 Cache Properties

CPUID Fn8000_001D_EBX_x0 reports topology information for the DC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:22	**CacheNumWays: cache number of ways**. Cache number of ways is CacheNumWays+1. Value: 007h.

21:12	**CachePhysPartitions: cache physical line partitions**. Value: 000h. Cache partitions is CachePhysPartitions+1.
11:0	**CacheLineSize: cache line size in bytes**. Value: 03Fh. Cache line size in bytes is CacheLineSize+1.

CPUID Fn8000_001D_EBX_x1 Cache Properties

CPUID Fn8000_001D_EBX_x1 reports topology information for the IC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:22	**CacheNumWays: cache number of ways**. See: CPUID Fn8000_001D_EBX_x0[CacheNumWays]. Value: 002h.
21:12	**CachePhysPartitions: cache physical line partitions**. Value: 000h. See: CPUID Fn8000_001D_EBX_x0[CachePhysPartitions].
11:0	**CacheLineSize: cache line size in bytes**. Value: 03Fh. See: CPUID Fn8000_001D_EBX_x0[CacheLineSize].

CPUID Fn8000_001D_EBX_x2 Cache Properties

CPUID Fn8000_001D_EBX_x2 reports topology information for the L2. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:22	**CacheNumWays: cache number of ways**. Value: 00Fh. See: CPUID Fn8000_001D_EBX_x0[CacheNumWays].
21:12	**CachePhysPartitions: cache physical line partitions**. Value: 000h. See: CPUID Fn8000_001D_EBX_x0[CachePhysPartitions].
11:0	**CacheLineSize: cache line size in bytes**. Value: 03Fh. See: CPUID Fn8000_001D_EBX_x0[CacheLineSize].

CPUID Fn8000_001D_EBX_x3 Cache Properties

CPUID Fn8000_001D_EAX_x3 reports done/null. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:22	**CacheNumWays: cache number of ways**. Value: 0. See: CPUID Fn8000_001D_EBX_x0[CacheNumWays].

21:12	**CachePhysPartitions: cache physical line partitions**. Value: 000h. See: CPUID Fn8000_001D_EBX_x0[CachePhysPartitions].
11:0	**CacheLineSize: cache line size in bytes**. Value: 0. See: CPUID Fn8000_001D_EBX_x0[CacheLineSize].

CPUID Fn8000_001D_ECX_x0 Cache Properties

CPUID Fn8000_001D_ECX_x0 reports topology information for the DC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:0	**CacheNumSets: cache number of sets**. Cache number of sets is CacheNumSets+1. Value: 0000_003Fh.

CPUID Fn8000_001D_ECX_x1 Cache Properties

CPUID Fn8000_001D_ECX_x1 reports topology information for the IC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:0	**CacheNumSets: cache number of sets**. See: CPUID Fn8000_001D_ECX_x0[CacheNumSets]. Value: 0000_01FFh.

CPUID Fn8000_001D_ECX_x2 Cache Properties

CPUID Fn8000_001D_ECX_x2 reports topology information for the L2. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:0	**CacheNumSets: cache number of sets**. Value: Product-specific. See: CPUID Fn8000_001D_ECX_x0[CacheNumSets].

CPUID Fn8000_001D_ECX_x3 Cache Properties

CPUID Fn8000_001D_EAX_x3 reports done/null. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:0	Reserved. Value: 0000_0000h.

CPUID Fn8000_001D_EDX_x0 Cache Properties

CPUID Fn8000_001D_EDX_x0 reports topology information for the DC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:2	Reserved.
1	**CacheInclusive: cache inclusive**. Value: 0. 0=Cache is not inclusive of lower cache levels. 1=Cache is inclusive of lower cache levels.
0	**WBINVD: Write-Back Invalidate/Invalidate**. Value: 0. 0=WBINVD/INVD invalidates all lower level caches of non-originating cores sharing this cache. 1=WBINVD/INVD not ensured to invalidate all lower level caches of non-originating cores sharing this cache.

CPUID Fn8000_001D_EDX_x1 Cache Properties

CPUID Fn8000_001D_EDX_x1 reports topology information for the IC. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:2	Reserved.
1	**CacheInclusive: cache inclusive**. Value: 0. See: CPUID Fn8000_001D_EDX_x0[CacheInclusive].
0	**WBINVD: Write-Back Invalidate/Invalidate**. Value: 0. See: CPUID Fn8000_001D_EDX_x0[WBINVD].

CPUID Fn8000_001D_EDX_x2 Cache Properties

CPUID Fn8000_001D_EDX_x2 reports topology information for the L2. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:2	Reserved.
1	**CacheInclusive: cache inclusive**. See: CPUID Fn8000_001D_EDX_x0[CacheInclusive]. Value: 0.
0	**WBINVD: Write-Back Invalidate/Invalidate**. Value: 1. See: CPUID Fn8000_001D_EDX_x0[WBINVD].

CPUID Fn8000_001D_EDX_x3 Cache Properties

CPUID Fn8000_001D_EAX_x3 reports done/null. See CPUID Fn8000_001D_EAX_x0.

Bits	Description
31:0	Reserved. Value: 0000_0000h.

CPUID Fn8000_001E_EAX Extended APIC ID

If CPUID Fn8000_0001_ECX[TopologyExtensions] == 0 then CPUID Fn8000_001E_E[D,C,B,A]X is reserved. If (MSR0000_001B[ApicEn] == 0) then CPUID Fn8000_001E_EAX[ExtendedApicId] is reserved.

Bits	Description
31:0	**ExtendedApicId: extended APIC ID**. Value: IF (MSR0000_001B[ApicEn] == 0) THEN 0000_0000h. ELSE APIC20[31:0]. ENDIF.

CPUID Fn8000_001E_EBX Compute Unit Identifiers

See CPUID Fn8000_001E_EAX.

Bits	Description
31:16	Reserved.
15:8	**ThreadsPerComputeUnit: threads per compute unit**. The number of threads per compute unit is ThreadsPerComputeUnit+1. Value: Product-specific.
7:0	**ComputeUnitId: compute unit ID**. Identifies the processor compute unit ID. Value: Product-specific.

CPUID Fn8000_001E_ECX Node Identifiers

See CPUID Fn8000_001E_EAX.

Bits	Description
31:0	Reserved.

CPUID Fn8000_001E_EDX Reserved

See CPUID Fn8000_001E_EAX.

Bits	Description
31:0	Reserved.

3.20 MSRs - MSR0000_xxxx

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. MSRs are accessed through x86 WRMSR and RDMSR instructions.

MSR0000_0000 Load-Store MCA Address

Bits	Description
63:0	Alias of MSR0000_0402.

MSR0000_0001 Load-Store MCA Status

Bits	Description
63:0	Alias of MSR0000_0401.

MSR0000_0010 Time Stamp Counter (TSC)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	**TSC[63:32]: time stamp counter high**. See: TSC[31:0].
31:0	**TSC[31:0]: time stamp counter low**. Read-write; Updated-by-hardware. TSC[63:0] = {TSC[63:32],TSC[31:0]}. The TSC increments at the P0 frequency. This field uses software P-state numbering. See 2.5.2.1.1.1 [Software P-state Numbering]. The TSC counts at the same rate in all P-states, all C states, S0, or S1. A read of this MSR in guest mode is affected by MSRC000_0104 [Time Stamp Counter Ratio (TscRateMsr)]. The value (TSC/TSCRatio) is the TSC P0 frequency based value (as if TSCRatio == 1.0) when (TSCRatio != 1.0).

MSR0000_001B APIC Base Address (APIC_BAR)

Bits	Description
63:40	MBZ.
39:12	**ApicBar[39:12]: APIC base address register**. Read-write. Reset: 00_FEE0_0h. Specifies the base address, physical address [39:12], for the APICXX register set in xAPIC mode. See 2.4.9.1.2 [APIC Register Space].
11	**ApicEn: APIC enable**. Read-write. Reset: 0. See 2.4.9.1.2 [APIC Register Space]. 1=Local APIC is enabled in xAPIC mode.
10	MBZ.
9	MBZ.
8	**BSC: boot strap core**. Read-write; Updated-by-hardware. Reset: X. 1=The core is the boot core of the BSP. 0=The core is not the boot core of the BSP.
7:0	MBZ.

MSR0000_002A Cluster ID (EBL_CR_POWERON)

Read; GP-write.

Bits	Description
63:18	MBZ.
17:16	**ClusterID**. Reset: 00b. The field does not affect hardware.
15:0	MBZ.

MSR0000_00E7 Max Performance Frequency Clock Count (MPERF)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**MPERF: maximum core clocks counter**. Read-write; Updated-by-hardware. Incremented by hardware at the P0 frequency while the core is in C0. This register does not increment when the core is in the stop-grant state. In combination with MSR0000_00E8, this is used to determine the effective frequency of the core. A read of this MSR in guest mode is affected by MSRC000_0104 [Time Stamp Counter Ratio (TscRateMsr)]. This field uses software P-state numbering. See MSRC001_0015[EffFreqCntMwait], 2.5.2.3 [Effective Frequency], and 2.5.2.1.1.1 [Software P-state Numbering].

MSR0000_00E8 Actual Performance Frequency Clock Count (APERF)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**APERF: actual core clocks counter**. Read-write; Updated-by-hardware. This register increments in proportion to the actual number of core clocks cycles while the core is in C0. The register does not increment when the core is in the stop-grant state. See MSR0000_00E7.

MSR0000_00FE MTRR Capabilities (MTRRcap)

Read; GP-write. Reset: 0000_0000_0000_0508h.

Bits	Description
63:11	Reserved.
10	**MtrrCapWc: write-combining memory type**. 1=The write combining memory type is supported.
9	Reserved.
8	**MtrrCapFix: fixed range register**. 1=Fixed MTRRs are supported.
7:0	**MtrrCapVCnt: variable range registers count**. Specifies the number of variable MTRRs supported.

MSR0000_0174 SYSENTER CS (SYSENTER_CS)

Bits	Description
63:32	RAZ.

31:16	Reserved.
15:0	**SysEnterCS: SYSENTER target CS**. Read-write. Reset: 0000h. Holds the called procedure code segment.

MSR0000_0175 SYSENTER ESP (SYSENTER_ESP)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:0	**SysEnterESP: SYSENTER target SP**. Read-write. Holds the called procedure stack pointer.

MSR0000_0176 SYSENTER EIP (SYSENTER_EIP)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:0	**SysEnterEIP: SYSENTER target IP**. Read-write. Holds the called procedure instruction pointer.

MSR0000_0179 Global Machine Check Capabilities (MCG_CAP)

Read; GP-write.

Bits	Description
63:9	Reserved.
8	**McgCtlP: MCG_CTL register present**. Value: 1. 1=The machine check control registers (MCi_CTL) are present. See 2.16.1 [Machine Check Architecture].
7:0	**Count**. Value: 07h. Indicates the number of error reporting banks visible to each core. 06h=Error-reporting banks 0 through 5. See 2.16.1.1 [Machine Check Registers].

MSR0000_017A Global Machine Check Status (MCG_STAT)

Reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture].

Bits	Description
63:3	Reserved.
2	**MCIP: machine check in progress**. Read-write; Set-by-hardware. 1=A machine check is in progress.
1	**EIPV: error instruction pointer valid**. Read-write; Updated-by-hardware. 1=The instruction pointer that was pushed onto the stack by the machine check mechanism references the instruction that caused the machine check error.
0	**RIPV: restart instruction pointer valid**. Read-write; Updated-by-hardware. 1=Program execution can be reliably restarted at the EIP address on the stack. 0=The interrupt was not precise and/or the process (task) context may be corrupt; continued operation of this process may not be possible without intervention, however system processing or other processes may be able to continue with appropriate software clean up.

MSR0000_017B Global Machine Check Exception Reporting Control (MCG_CTL)

Read-write. Reset: 0000_0000_0000_0000h. This registers controls enablement of the individual error reporting banks; see 2.16.1 [Machine Check Architecture]. When a machine check register bank is not enabled in MCG_CTL, errors for that bank are not logged or reported, and actions enabled through the MCA are not taken; each MCi_CTL register identifies which errors are still corrected when MCG_CTL[i] is disabled.

Bits	Description
63:7	Unused.
6	**MC6En: MC6 register bank enable**. 1=The MC6 machine check register bank is enabled.
5	**MC5En: MC5 register bank enable**. 1=The MC5machine check register bank is enabled.
4	**MC4En: MC4 register bank enable**. 1=The MC4 machine check register bank is enabled for all cores of the node.
3	Unused.
2	**MC2En: MC2 register bank enable**. 1=The MC2 machine check register bank is enabled.
1	**MC1En: MC1 register bank enable**. 1=The MC1 machine check register bank is enabled.
0	**MC0En: MC0 register bank enable**. 1=The MC0 machine check register bank is enabled.

MSR0000_01D9 Debug Control (DBG_CTL_MSR)

Bits	Description
63:7	Reserved.
6	MBZ.
5:2	**PB: performance monitor pin control**. Read-write. Reset: 0. This field does not control any hardware.
1	**BTF**. Read-write. Reset: 0. 1=Enable branch single step.
0	**LBR**. Read-write. Reset: 0. 1=Enable last branch record.

MSR0000_01DB Last Branch From IP (BR_FROM)

Read; GP-write; Not-same-for-all; Updated-by-hardware. Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**LastBranchFromIP**. Loaded with the segment offset of the branch instruction.

MSR0000_01DC Last Branch To IP (BR_TO)

Read; GP-write; Not-same-for-all; Updated-by-hardware. Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**LastBranchToIP**. Holds the target RIP of the last branch that occurred before an exception or interrupt.

MSR0000_01DD Last Exception From IP

Read; GP-write; Not-same-for-all; Updated-by-hardware. Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**LastIntFromIP**. Holds the source RIP of the last branch that occurred before the exception or interrupt.

MSR0000_01DE Last Exception To IP

Read; GP-write; Not-same-for-all; Updated-by-hardware. Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**LastIntToIP**. Holds the target RIP of the last branch that occurred before the exception or interrupt.

MSR0000_020[F:0] Variable-Size MTRRs Base/Mask

Per-compute-unit.
Each MTRR (MSR0000_020[F:0] [Variable-Size MTRRs Base/Mask], MSR0000_02[6F:68,59:58,50], or MSR0000_02FF [MTRR Default Memory Type (MTRRdefType)]) specifies a physical address range and a corresponding memory type (MemType) associated with that range. Setting the memory type to an unsupported value results in a #GP.

The variable-size MTRRs come in pairs of base and mask registers (MSR0000_0200 and MSR0000_0201 are the first pair, etc.). Variables MTRRs are enabled through MSR0000_02FF[MtrrDefTypeEn]. A core access--with address CPUAddr--is determined to be within the address range of a variable-size MTRR if the following equation is true:
CPUAddr39:12] & PhyMask[39:12] == PhyBase[39:12] & PhyMask[39:12].

For example, if the variable MTRR spans 256 KB and starts at the 1 MB address. The PhyBase would be set to 0_0010_0000h and the PhyMask to F_FFFC_0000h (with zeros filling in for bits[11:0]). This results in a range from 0_0010_0000h to 0_0013_FFFF.

MSR0000_020[E,C,A,8,6,4,2,0] Variable-Size MTRRs Base

Table 234: Register Mapping for MSR0000_020[E,C,A,8,6,4,2,0]

Register	Function
MSR0000_0200	Range 0
MSR0000_0202	Range 1
MSR0000_0204	Range 2
MSR0000_0206	Range 3
MSR0000_0208	Range 4
MSR0000_020A	Range 5
MSR0000_020C	Range 6
MSR0000_020E	Range 7

Table 235: Valid Values for Memory Type Definition

Bits	Description
000b	UC or uncacheable.
001b	WC or write combining.
011b-010b	Reserved
100b	WT or write through.
101b	WP or write protect.
110b	WB or write back.
111b	Reserved

Bits	Description
63:48	MBZ.
47:12	**PhyBase: base address**. Read-write. Reset: 0.
11:3	MBZ.
2:0	**MemType: memory type**. Read-write. Reset: 0. Address range from 00000h to 0FFFFh. See: Table 235 [Valid Values for Memory Type Definition].

MSR0000_020[F,D,B,9,7,5,3,1] Variable-Size MTRRs Mask

Table 236: Register Mapping for MSR0000_020[F,D,B,9,7,5,3,1]

Register	Function
MSR0000_0201	Range 0
MSR0000_0203	Range 1
MSR0000_0205	Range 2
MSR0000_0207	Range 3
MSR0000_0209	Range 4
MSR0000_020B	Range 5
MSR0000_020D	Range 6
MSR0000_020F	Range 7

Bits	Description
63:48	MBZ.
47:12	**PhyMask: address mask**. Read-write. Reset: 0.
11	**Valid: valid**. Read-write. Reset: 0. 1=The variable-size MTRR pair is enabled.
10:0	MBZ.

MSR0000_02[6F:68,59:58,50] Fixed-Size MTRRs

Per-compute-unit.
See MSR0000_020[F:0] for general MTRR information. Fixed MTRRs are enabled through MSR0000_02FF[MtrrDefTypeFixEn, MtrrDefTypeEn]. For addresses below 1MB, the appropriate Fixed MTRRs override the default access destination. Each fixed MTRR includes two bits, RdDram and WrDram, that determine the destination based on the access type.
See 2.4.6.1.2 [Determining The Access Destination for Core Accesses].

Table 237: Register Mapping for MSR0000_02[6F:68,59:58,50]

Register	Function
MSR0000_0250	64K Range
MSR0000_0258	16K_0 Range
MSR0000_0259	16K_1 Range
MSR0000_0268	4K_0 Range
MSR0000_0269	4K_1 Range
MSR0000_026A	4K_2 Range
MSR0000_026B	4K_3 Range
MSR0000_026C	4K_4 Range
MSR0000_026D	4K_5 Range
MSR0000_026E	4K_6 Range
MSR0000_026F	4K_7 Range

Table 238: Field Mapping for MSR0000_02[6F:68,59:58,50]

Register	Bits							
	63:56	55:48	47:40	39:32	31:24	23:16	15:8	7:0
MSR0000_0250	64K_70000	64K_60000	64K_50000	64K_40000	64K_30000	64K_20000	64K_10000	64K_00000
MSR0000_0258	16K_9C000	16K_98000	16K_94000	16K_90000	16K_8C000	16K_88000	16K_84000	16K_80000
MSR0000_0259	16K_BC000	16K_B8000	16K_B4000	16K_B0000	16K_AC000	16K_A8000	16K_A4000	16K_A0000
MSR0000_0268	4K_C7000	4K_C6000	4K_C5000	4K_C4000	4K_C3000	4K_C2000	4K_C1000	4K_C0000
MSR0000_0269	4K_CF000	4K_CE000	4K_CD000	4K_CC000	4K_CB000	4K_CA000	4K_C9000	4K_C8000
MSR0000_026A	4K_D7000	4K_D6000	4K_D5000	4K_D4000	4K_D3000	4K_D2000	4K_D1000	4K_D0000
MSR0000_026B	4K_DF000	4K_DE000	4K_DD000	4K_DC000	4K_DB000	4K_DA000	4K_D9000	4K_D8000
MSR0000_026C	4K_E7000	4K_E6000	4K_E5000	4K_E4000	4K_E3000	4K_E2000	4K_E1000	4K_E0000
MSR0000_026D	4K_EF000	4K_EE000	4K_ED000	4K_EC000	4K_EB000	4K_EA000	4K_E9000	4K_E8000
MSR0000_026E	4K_F7000	4K_F6000	4K_F5000	4K_F4000	4K_F3000	4K_F2000	4K_F1000	4K_F0000
MSR0000_026F	4K_FF000	4K_FE000	4K_FD000	4K_FC000	4K_FB000	4K_FA000	4K_F9000	4K_F8000

Bits	Description
63:61	MBZ.
60	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
59	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
58:56	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].

55:53	MBZ.
52	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
51	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
50:48	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].
47:45	MBZ.
44	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
43	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
42:40	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].
39:37	MBZ.
36	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
35	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
34:32	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].
31:29	MBZ.
28	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
27	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
26:24	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].
23:21	MBZ.
20	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
19	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
18:16	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].
15:13	MBZ.
12	**RdDram: read DRAM**. See: MSR0000_02[6F:68,59:58,50][4].
11	**WrDram: write DRAM**. See: MSR0000_02[6F:68,59:58,50][3].
10:8	**MemType: memory type**. See: MSR0000_02[6F:68,59:58,50][2:0].
7:5	MBZ.
4	**RdDram: read DRAM**. IF (MSRC001_0010[MtrrFixDramModEn]) THEN Read-write. ELSE MBZ. ENDIF. Reset: 0. 0=Read accesses to the range are marked as MMIO. 1=Read accesses to the range are marked as destined for DRAM. Address range from 00000h to 0FFFFh. See: MSRC001_0010[MtrrFixDramEn, MtrrFixDramModEn]).
3	**WrDram: write DRAM**. IF (MSRC001_0010[MtrrFixDramModEn]) THEN Read-write. ELSE MBZ. ENDIF. Reset: 0. 0=Write accesses to the range are marked as MMIO. 1=Write accesses to the range are marked as destined for DRAM. Address range from 00000h to 0FFFFh. See: MSRC001_0010[MtrrFixDramEn, MtrrFixDramModEn]).
2:0	**MemType: memory type**. Read-write. Reset: 0. Address range from 00000h to 0FFFFh. See: Table 235 [Valid Values for Memory Type Definition].

MSR0000_0277 Page Attribute Table (PAT)

This register specifies the memory type based on the PAT, PCD, and PWT bits in the virtual address page tables.

Bits	Description
63:59	MBZ.
58:56	**PA7MemType**. See: PA0MemType. Reset: 0h. Default UC. MemType for {PAT, PCD, PWT} = 7h.
55:51	MBZ.
50:48	**PA6MemType**. See: PA0MemType. Reset: 7h. Default UC-. MemType for {PAT, PCD, PWT} = 6h.
47:43	MBZ.
42:40	**PA5MemType**. See: PA0MemType. Reset: 4h. Default WT. MemType for {PAT, PCD, PWT} = 5h.
39:35	MBZ.
34:32	**PA4MemType**. See: PA0MemType. Reset: 6h. Default WB. MemType for {PAT, PCD, PWT} = 4h.
31:27	MBZ.
26:24	**PA3MemType**. See: PA0MemType. Reset: 0h. Default UC. MemType for {PAT, PCD, PWT} = 3h.
23:19	MBZ.
18:16	**PA2MemType**. See: PA0MemType. Reset: 7h. Default UC-. MemType for {PAT, PCD, PWT} = 2h.
15:11	MBZ.
10:8	**PA1MemType**. See: PA0MemType. Reset: 4h. Default WT. MemType for {PAT, PCD, PWT} = 1h.
7:3	MBZ.
2:0	**PA0MemType**. Read-write. Reset: 6h. MemType for {PAT, PCD, PWT} = 0h. Bits / Description / Bits / Description 0h UC or uncacheable. 4h WT or write through. 1h WC or write combining. 5h WP or write protect. 2h MBZ. 6h WB or write back. 3h MBZ. 7h UC- or uncacheable (overridden by WC state).

MSR0000_02FF MTRR Default Memory Type (MTRRdefType)

Per-compute-unit.
See MSR0000_020[F:0] for general MTRR information.

Bits	Description
63:12	MBZ.
11	**MtrrDefTypeEn: variable and fixed MTRR enable**. Read-write. Reset: 0. 1=MSR0000_020[F:0] [Variable-Size MTRRs Base/Mask], and MSR0000_02[6F:68,59:58,50] [Fixed-Size MTRRs] are enabled. 0=Fixed and variable MTRRs are not enabled.
10	**MtrrDefTypeFixEn: fixed MTRR enable**. Read-write. Reset: 0. 1=MSR0000_02[6F:68,59:58,50] [Fixed-Size MTRRs]are enabled. This field is ignored (and the fixed MTRRs are not enabled) if MSR0000_02FF[MtrrDefTypeEn] == 0.

9:8	MBZ.
7:0	**MemType: memory type**. Read-write. Reset: 0. If MtrrDefTypeEn == 1 then MemType specifies the memory type for memory space that is not specified by either the fixed or variable range MTRRs. If MtrrDefTypeEn == 0 then the default memory type for all of memory is UC. Valid encodings are {00000b, MSR0000_02[6F:68,59:58,50][2:0]}.

MSR0000_0400 MC0 Machine Check Control (MC0_CTL)

Read-write. Reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSRC001_0044 [DC Machine Check Control Mask (MC0_CTL_MASK)].

Bits	Description
63:12	Unused.
11	**MAB: Miss Address Buffer parity**.
10	**HWA: hardware assertion**.
9	**IntErrType1: internal error type 1**.
8	**IntErrType2: internal error type 2**.
7	**SRDE: read data error**. System read data errors on cache fill.
6	**LFE: line fill error**. Uncorrectable error on cache fill.
5	**SCBP: SCB parity**.
4	**SQP: store queue parity**.
3	**LQP: load queue parity**.
2	**DatP: data parity**.
1	**TLBP: TLB parity**.
0	**TagP: tag parity**.

MSR0000_0401 MC0 Machine Check Status (MC0_STATUS)

See 2.16.1 [Machine Check Architecture]. See MSRC001_0015[McStatusWrEn]. Table 239 describes each error type. Table 240 describes the error codes and status register settings for each error type. MSR0000_0001 is an alias of MSR0000_0401.

Bits	Description
63	**Val: error valid**. Read-write; Set-by-hardware. Cold reset: 0. 1=This bit indicates that a valid error has been detected. This bit should be cleared to 0 by software after the register has been read.

62	**Overflow: error overflow**. Read-write; Set-by-hardware. Cold reset: 0. 1=An error was detected while the valid bit (Val) was set; at least one error was not logged. Overflow is set independently of whether the existing error is overwritten. The following hierarchy identifies the error logging priorities. 1. Uncorrectable errors 2. Deferred errors 3. Correctable errors The machine check mechanism handles the contents of MC*i*_STATUS during overflow as follows: • Higher priority errors overwrite lower priority errors. • New errors of equal or lower priority do not overwrite existing errors. • Uncorrectable errors which are not logged due to overflow result in setting PCC, unless the new uncorrectable error is of the same type and in the same reportable address range as the existing error.
61	**UC: error uncorrected**. Read-write; Updated-by-hardware. Cold reset: 0. 1=The error was not corrected by hardware.
60	**En: error enable**. Read-write; Updated-by-hardware. Cold reset: 0. 1=MCA error reporting is enabled for this error, as indicated by MCi_CTL.
59	**MiscV: miscellaneous error register valid**. Read-write; Updated-by-hardware. Cold reset: 0. 1=Valid thresholding in MSR0000_0403.
58	**AddrV: error address valid**. Read-write; Updated-by-hardware. Cold reset: 0. 1=MCi_ADDR contains address information associated with the error.
57	**PCC: processor context corrupt**. Read-write; Updated-by-hardware. Cold reset: 0. 1=Hardware context held by the processor may have been corrupted. Continued operation of the system may have unpredictable results. The error is not recoverable or survivable, and the system should be reinitialized. See 2.16.1.6.1 [Differentiation Between System-Fatal and Process-Fatal Errors].
56:45	Reserved.
44	**Deferred: deferred error**. Read-write; Updated-by-hardware. Cold reset: 0. 1=A deferred error was created. A deferred error is the result of an uncorrectable data error which did not immediately cause a processor exception; the data is poisoned and an exception is deferred until the data is loaded by a core.
43	**Poison: poison error**. Read-write; Updated-by-hardware. Cold reset: 0. 1=The error was the result of attempting to consume poisoned data. This indicator does not apply to MSR0000_0411 [MC4 Machine Check Status (MC4_STATUS)].
42:40	Reserved.
39:36	**Way: cache way in error.** Read-write; Updated-by-hardware. Cold reset: 0. Indicates the cache way in error. Bits Description 0h Way 0 1h Way 1 2h Way 2 3h Way 3 Fh-4h Reserved.

35:22	Reserved.
21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware. Cold reset: 0. Logs an extended error code when an error is detected. This model-specific field is used in conjunction with ErrorCode to identify the error sub-type for root cause analysis (see 2.16.1.5 [Error Code]). See Table 240 for expected values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware. Cold reset: 0. See 2.16.1.5 [Error Code] for details on decoding this field. See Table 240 for expected values.

Table 239: MC0 Error Descriptions

Error Type	Error Sub-type	Description[1]	CTL[2]	EAC[3]
Line Fill Error	-	An uncorrectable error occurred during a line fill from the L2 cache or the NB. (Note: For IO read, may not actually install to L1 cache.)	LineFillError	E
Data Cache Error	Data array	Error occurred in cache data array access.	DatP	D
	SCB	Error occurred in SCB access.	SCBP	D
	STQ	Error occurred in STQ access.	SQP	D
Tag Error	Tag array	A tag error was encountered. If uncorrectable, this errors is system fatal and results in a sync flood.	TagP	D
	STQ	Error occurred in STQ access.	SQP	D
	LDQ	Error occurred in LDQ access.	LQP	D
L1 TLB Error	TLB parity	Parity error in L1 TLB access.	TLBP	D
	TLB multimatch	Lookup hit on multiple entries.		D
	Locked TLB miss	TLB miss occurred after lock granted.		E
System Read Data Error	-	An error occurred during an attempted read of data from the NB. Possible reasons include master abort, target abort.	SRDE	E
Internal Error	IntErrType1	An internal error condition was detected which prohibits the core from continuing execution.	IntErrType1	E
	IntErrType2		IntErrType2	E
Internal Error	Hardware Assert	A hardware assertion was detected.	HWA	D
MAB Parity	Miss Address Buffer Parity	A parity error occurred in the Miss Address Buffer.	MAB	D

1. CID: core ID. All LS errors are reported to the affected core; see 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
2. See MSR0000_0400.
3. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled. See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].

Table 240: MC0 Error Signatures

Error Type	Error	ErrorCodeExt	Error Code						UC	ADDRV	PCC	Deferred	Poison
			Type	UU/PP	T	RRRR	II/TT	LL					
Line Fill Error	-	01h	MEM	-	-	DRD	D	LG	1	1	0	-0	-1
Data Cache Error	Data array	00h				DRD	D	L1	0/1	1	0	-0	-0
	SCB	03h											
	STQ	02h											
Data Cache Error	MAB	10h	MEM	-	-	DRD	G	L1	1	0	0	0	0
Tag Error	Tag array	10h	MEM	- SRC	- 0	DRD, DWR, Probe	G	L1	1	0/1	1	-0	-0
	STQ	11h				DWR			0/1	1	0/1		
	LDQ	12h				DRD			0/1	1	0		
L1 TLB Error	TLB parity	00h	TLB			-	D	L1	0/1	1	0	-0	-0
	TLB Multimatch	01h							0/1	0			
	Locked TLB miss	02h							1	1			
System Read Data Error	-	00h	BUS			DRD	MEM/IO	LG	1	1	0	-0	-0
Internal Error	Type1	01h		GEN	1	GEN	GEN	LG	1	0	0	-0	-0
	Type2	02h											
Internal Error	Hardware Assert	00-1Fh	INT	HWA	0	-	-	-	1	0	1	-0	-0

MSR0000_0402 MC0 Machine Check Address (MC0_ADDR)

Read-write; Updated-by-hardware. Cold reset: 0000_0000_0000_0000h. The MCi_ADDR register contains valid data if indicated by MCi_STATUS[AddrV]. See 2.16.1 [Machine Check Architecture]. MSR0000_0000 is an alias of MSR0000_0402.

Bits	Description
63:0	**ADDR: Address**. See Table 241.

Table 241: MC0 Address Register

Error Type	Error Sub-type	Bits	Description
Line Fill Error	-	63:48	Reserved
		47:6	**PhysAddr[47:6]**.
		5:0	Reserved
Data Cache Error	Data array	63:48	Reserved
		47:4	**PhysAddr[47:4]**.
		3:0	Reserved
Data Cache Error	SCB	63:12	Reserved
		11:4	**PhysAddr[11:4]**.
		3:0	Reserved

Table 241: MC0 Address Register (Continued)

Error Type	Error Sub-type	Bits	Description
Data Cache Error	STQ	63:5	Reserved
		4:0	**Index**.
Tag Error	Tag array	63:48	Reserved
		47:6	**PhysAddr[47:6]**.
		5:4	**PhysAddr[5:4]**. Not valid for probe errors.
		3:0	Reserved
	STQ	63:5	Reserved
		4:0	**Index**.
	LDQ	63:6	Reserved
		5:0	**Index**.
L1 TLB Error	TLB parity	63:48	Reserved
		47:12	**LinAddr[47:12]**.
		11:5	Reserved
		4:0	**TlbIndex**.
	Locked TLB miss	63:48	Reserved
		47:12	**LinAddr[47:12]**.
		11:0	Reserved
System Read Data Error	-	63:48	Reserved
		47:6	**PhysAddr[47:6]**.
		5:0	Reserved

MSR0000_0403 MC0 Machine Check Miscellaneous (MC0_MISC)

See 2.16.1.7 [Error Thresholding].

Bits	Description
63	**Valid**. IF (MSRC001_0015[McStatusWrEn]) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. 1=A valid CntP field is present in this register.
62	**CntP: counter present**. IF (MSRC001_0015[McStatusWrEn]) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. 1=A valid threshold counter is present.
61	**Locked**. IF (MSRC001_0015[McStatusWrEn]) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. 1=Writes to this register are ignored. This bit is set by BIOS to indicate that this register is not available for OS use. BIOS should set this bit if IntType is set to SMI.
60	**IntP: Interrupt support present**. IF (MSRC001_0015[McStatusWrEn] \| ~Locked) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. 1=IntType can be used to generate interrupts. 0=IntType and interrupt generation are not suported.
59:56	Reserved.

55:52	**LvtOffset: LVT offset**. IF (MSRC001_0015[McStatusWrEn] \| ~Locked) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0000b. BIOS: 1. Specifies the address of the LVT entry in the APIC registers as follows: LVT address = (LvtOffset shifted left 4 bits) + 500h (see APIC[530:500]). Only values 0 through 3 are valid; all others reserved.
51	**CntEn: counter enable**. Read-write; Updated-by-hardware. Reset: 0. 1=Count thresholding errors. See 2.16.1.7 [Error Thresholding].
50:49	**IntType: interrupt type**. Read-write. Cold reset: 0. Specifies the type of interrupt signaled when Ovrflw is set and IntP == 1. Bits Description 00b No Interrupt. 01b APIC based interrupt (see LvtOffset above) to all cores. 10b SMI trigger event (always routed to CpuCoreNum 0, as defined in 2.4.4 [Processor Cores and Downcoring]); see 2.4.10.2.3 [SMI Sources And Delivery]. 11b Reserved.
48	**Ovrflw: overflow**. Read-write; Set-by-hardware. Cold reset: 0. Set by hardware when ErrCnt transitions from FFEh to FFFh. When this field is set, ErrCnt no longer increments. When this bit is set and the IntP field == 1, the interrupt selected by the IntType field is generated.
47:44	Reserved.
43:32	**ErrCnt: error counter**. Read-write; Updated-by-hardware. Cold reset: 0. This is written by software to set the starting value of the error counter. This is incremented by hardware when errors are logged. When this counter overflows, it stays at FFFh (no rollover). The threshold value, written by software, is (FFFh - the desired error count (the number of errors necessary in order for an interrupt to be taken)); the desired error count of 0 (a write value of FFFh) is not supported.
31:24	**BlkPtr: Block pointer for additional MISC registers**. Read-only. Value: 00h. 00h=Extended MISC MSR block is not valid.
23:0	Reserved.

MSR0000_0404 MC1 Machine Check Control (MC1_CTL)

Per-compute-unit; Read-write.
Reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture].

Bits	Description
63:26	Unused.
25	**HWA: Hardware assertion**.
24	**DEDISPQ: decoder dispatch micro-op queue parity error**.
23	**IVP: IC valid bit parity error**.
22	**L1TLBM: IC L1 TLB multi-match error**.
21	**L2TLBM: IC L2 TLB multi-match error**.
20	**DFIFOE: decoder FIFO parity error**.
19	**DPDBE: decoder predecode buffer parity error**.
18	**DEIBP: decoder instruction buffer parity error**.
17	**DEUOPQP: Decoder micro-op queue parity error**.
16	**DEPRP: microcode patch buffer parity error**.
15	**BSRP: branch status register parity error**.

14	Unused.
13	**PQP: prediction queue parity error**.
12	**PFBP: prefetch buffer parity**.
11:10	Unused.
9	**SRDE: system read data error**.
8	Unused.
7	**LFE: line fill error**. Uncorrectable error on cache line fill.
6	**L1TP: L1 TLB parity error**.
5	**L2TP: L2 TLB parity error**.
4	**ISTP: L1 cache probe tag array parity error**.
3	**IMTP: L1 cache main tag array parity error**.
2	**IDP: L1 cache data array parity errors**.
1	Unused.
0	Unused.

MSR0000_0405 MC1 Machine Check Status (MC1_STATUS)

See 2.16.1 [Machine Check Architecture]. See MSRC001_0015[McStatusWrEn]. Table 242 describes each error type. Table 243 describes the error codes and status register settings for each error type.

Bits	Description
63	**Val: error valid**. See: MSR0000_0401[Val].
62	**Overflow: error overflow**. See: MSR0000_0401[Overflow].
61	**UC: error uncorrected**. See: MSR0000_0401[UC].
60	**En: error enable**. See: MSR0000_0401[En].
59	**MiscV: miscellaneous error register valid**. Read-write; Updated-by-hardware. Cold reset: 0. See: MSR0000_0401[MiscV]. 1=Valid thresholding in MSR0000_0407.
58	**AddrV: error address valid**. See: MSR0000_0401[AddrV].
57	**PCC: processor context corrupt**. See: MSR0000_0401[PCC].
56:45	Reserved.
44	**Deferred: deferred error**. See: MSR0000_0401[Deferred].
43	**Poison: poison error**. See: MSR0000_0401[Poison].
42:40	Reserved.
39:36	**Way: cache way in error**. Read-write; Updated-by-hardware. Cold reset: 0. Indicates the cache way in error. Bits Description 0h Way 0 1h Way 1 2h Way 2 Fh-3h Reserved.
35:22	Reserved.

21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware. Cold reset: 0. See MSR0000_0401[ErrorCodeExt]. See Table 243 for expected values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware. Cold reset: 0. See 2.16.1.5 [Error Code] for details on decoding this field. See Table 243 for expected values.

Table 242: MC1 Error Descriptions

Error Type	Error	Description	CTL[3]	CID[2]	EAC[1]
Line Fill Error	-		LineFill Poison	A	E
Instruction cache read error	IC Data Load Parity	A parity error occurred during load of data from the instruction cache. The data is discarded from the IC and can be refetched.	IDP	A	D
	IC valid bit	Parity error for IC valid bit.	IVP	A	D
	Main tag	A main tag parity error occurred.	IMTP	A	D
	Prediction queue	Parity error in prediction queue.	PQP	A	E
	PFB data/address	PFB data/address had a parity error. A PFB valid bit error, PFB multimatch error, Line Fill Error, or ReadData Error may additionally cause a PFB data/address error.	PFBP	A	E
	PFB valid bit	PFB valid bit had a parity error. This error may cause subsequent errors related to the entry, but the effect can be contained to the running process.		B	E
	PFB non-cacheable bit	PFB non-cacheable bit had a parity error.		B	E
	PFB promotion address error	An address parity error was detected when promoting from the PFB to the IC.		B	E
	Branch status register	A parity error was discovered in the branch status register. This error is uncorrectable, but the effect can be contained to the running process.	BSRP	A	E
Instruction cache read error	Microcode Patch Buffer	Parity error in the microcode patch buffer. This error is uncorrectable. If a reset is not performed or the patch area is not reloaded, then it is recommended that the compute unit be removed from the running configuration by the operating system if possible. After a reset, BIST is used to determine whether there is a hard fault in the RAM. If a hard fault is not found, the error was likely a transient upset and the RAM is not broken. This error can also be caused by an error in the microcode patch region of the CC6 save area if ECC is not enabled.	DEPRP	A	E

Table 242: MC1 Error Descriptions (Continued)

Error Type	Error	Description	CTL[3]	CID[2]	EAC[1]
Instruction cache read error	Decoder micro-op queue	Parity error in decode unit. This error is correctable unless the operation is for a non-cacheable operand.	DEUQ	A	E
	Decoder instruction buffer		DEIBP	A	E
	Decoder pre-decode buffer		DEPD	A	E
	Decoder fetch address FIFO		DEFF	A	E
Instruction cache read error	Decoder dispatch micro-op queue	Parity error in decode unit dispatch micro-op queue. This error is correctable unless the operation is for a non-cacheable operand.	DEDISPQ	A	E
Tag Probe	Probe tag error	A tag error was encountered during probe or victimization.	ISTP	0	D
	Probe tag valid bit	Parity error for IC probe tag valid bit.	IVP	0	D
L1 TLB	Parity	Parity error in L1 TLB.	L1TP	A	D
	Multimatch	Hit multiple entries in L1 TLB.	L1TLBM	A	D
L2 TLB	Parity	Parity error in L2 TLB.	L2TP	A	D
	Multimatch	Hit multiple entries in L2 TLB.	L2TLBM	A	D
System Read Data Error	-	An error occurred during an attempted demand read of data from the NB. Possible reasons include master abort, target abort.	SRDE	A	E
Internal Error	Hardware Assertion	Ahardware assertion was detected.	HWA	0	D

1. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled.See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
2. CID: core ID. A=Error reported to the affected core. 0=Error reported to core 0 of the Compute Unit. B=Error reported to all cores of the Compute Unit. See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
3. See MSR0000_0404.

Table 243: MC1 Error Signatures

Error Type	Error	ErrorCodeExt	Error Code						UC	ADDRV	PCC	Deferred	Poison
			Type	UU/PP	TT	RRRR	II/TT	LL					
Line Fill Error	-	00h	MEM	-	-	IRD	I	L2	1	1	0	-0	-1
Instruction Cache Read Error	IC data load parity	01h		-	-	IRD	I	L1	0	1	0	-0	-0
	IC valid bit	02h							0	1	0	-0	-0
	Main tag	03h							0	1	0	-0	-0
	Prediction queue	04h							1	0	0	-0	-0
	PFB data/address	05h							0/1	0	0	-0	-0
	PFB valid bit	0Dh							1	0	0	-0	-0
	PFB non-cacheable bit	0Ah							0/1	0	0	-0	-0
	PFB promotion address error	07h							1	0	1	-0	-0
	Branch status register	06h							1	0	0	-0	-0
	Microcode Patch Buffer	10h						LG	1	1	1	-0	-0
	Decoder micro-op queue	11h						L1	0/1	1	0	-0	-0
	Decoder instruction buffer	12h							0/1	1	0	-0	-0
	Decoder pre-decode buffer	13h							0/1	0	0	-0	-0
	Decoder fetch address FIFO	14h							0/1	1	0	-0	-0
Instruction cache read error	Decoder dispatch micro-op queue	15h	MEM	-	-	IRD	I	L1	0/1	1	0	-0	-0
Tag Probe	Probe tag error	08h	MEM	-	-	Probe	I	L1	0	1	0	-0	-0
	Probe tag valid bit	09h											
L1 TLB	Parity	00h	TLB	-	-	-	I	L1	0	1	0	-0	-0
	Multimatch	01h											
L2 TLB	Parity	00h						L2					
	Multimatch	01h											
System Read Data Error	-	00h	BUS	SRC	0	IRD	MEM	LG	1	1	0	-0	-0
Internal Error	Hardware Assertion	00h-3Fh	INT	HWA	0	-	-	-	1	0	1	0	0

MSR0000_0406 MC1 Machine Check Address (MC1_ADDR)

Read-write; Updated-by-hardware. Cold reset: 0000_0000_0000_0000h. The MCi_ADDR register contains valid data if indicated by MCi_STATUS[AddrV]. See 2.16.1 [Machine Check Architecture].

Bits	Description
63:0	**ADDR: Address**. See Table 244.

The following table defines the address register as a function of error type.

Table 244: MC1 Address Register

Error Type	Error Sub-Type	Bits	Description
Line Fill Error	-	63:48	Reserved
		47:6	**LinAddr[47:6]**.
		5:0	Reserved
Instruction cache read error	IC data load parity	63:48	Reserved
		47:3	**LinAddr[47:3]**. [7:6] Description 00b Bank 0 01b Bank 1 10b Bank 2 11b Bank 3
		2:1	**Way**. Bits Description 00b Way 0 01b Way 1 10b Way 2 11b Reserved
		0	Reserved
Instruction cache read error	IC valid bit	63:48	Reserved
		47:6	**LinAddr[47:6]**. [7:6] Description 00b Bank 0 01b Bank 1 10b Bank 2 11b Bank 3
		5:0	Reserved

Table 244: MC1 Address Register (Continued)

Error Type	Error Sub-Type	Bits	Description
Instruction cache read error	Main tag	63:48	Reserved
		47:6	**LinAddr[47:6]**. [7:6] Description 00b Bank 0 01b Bank 1 10b Bank 2 11b Bank 3
		5:3	Reserved
		2:1	**Way**. Bits Description 00b Way 0 01b Way 1 10b Way 2 11b Reserved
		1:0	Reserved
Instruction Cache Read Error	Microcode Patch Buffer	63:4	Reserved
		3:0	Line group index.
Instruction cache read error	Decoder micro-op queue	63:2	Reserved
		1:0	Micro-op queue slot in error.
Instruction cache read error	Decoder instruction buffer	63:3	Reserved
		2	**PrefixMaskMismatch**. If (PrefixMaskMismatch == 1) then BankAndParityBitInError = 00b.
		1:0	**BankAndParityBitInError**. Bits Description 00b Bank A, parity bit 0 or 1 01b Bank B, parity bit 0 or 1 10b Bank A, parity bit 2 or 3 11b Bank B, parity bit 2 or 3
Instruction cache read error	Decoder fetch address FIFO	63:2	Reserved
		1	**BsrTagParityError**.
		0	**BankInError**. 0=Bank A. 1=Bank B.
Instruction cache read error	Decoder dispatch micro-op queue	63:2	Reserved
		1:0	Micro-op queue slot in error.

Table 244: MC1 Address Register (Continued)

Error Type	Error Sub-Type	Bits	Description
Tag Probe	Probe tag error	63:50	Reserved
		49:48	**BankBitmask**. Bit Description [0] Bank 6 [1] Bank 7
		47:6	**PhysAddr[47:6]**.
		5:0	**BankBitmask**. Bit Description [0] Bank 0 [1] Bank 1 [2] Bank 2 [3] Bank 3 [4] Bank 4 [5] Bank 5
Tag Probe	Probe tag valid bit	63:48	Reserved
		47:6	**PhysAddr[47:6]**.
		5:0	Reserved
L1 TLB	Parity, Multimatch	63:48	Reserved
		47:12	**LinAddr[47:12]**. 4-KB page: • [47:12]: LinAddr[47:12]. 2-MB page: • [47:20]: LinAddr[47:20]. • [19:12]: Reserved
		11:3	Reserved
		2:0	**BankBitmask**. Bit Description [0] Bank 0 [1] Bank 1 [2] Bank 2
L2 TLB	Parity, Multimatch	63:48	Reserved
		47:12	**LinAddr[47:12]**. (4-KB page size only)
		11:4	Reserved
		3:0	**MatchLines**.

MSR0000_0407 MC1 Machine Check Miscellaneous (MC1_MISC)

Cold reset: 0000_0000_0000_0000h.
See 2.16.1.7 [Error Thresholding].

Bits	Description
63	**Valid**. See: MSR0000_0403[Valid].
62	**CntP: counter present**. See: MSR0000_0403[CntP].

61	**Locked**. See: MSR0000_0403[Locked].
60	**IntP: Interrupt support present.** See: MSR0000_0403[IntP].
59:56	Reserved.
55:52	**LvtOffset: LVT offset**. See: MSR0000_0403[LvtOffset].
51	**CntEn: counter enable**. See: MSR0000_0403[CntEn].
50:49	**IntType: interrupt type**. See: MSR0000_0403[IntType].
48	**Ovrflw: overflow.** See: MSR0000_0403[Ovrflw].
47:44	Reserved.
43:32	**ErrCnt: error counter**. See: MSR0000_0403[ErrCnt].
31:24	**BlkPtr: Block pointer for additional MISC registers**. See: MSR0000_0403[BlkPtr].
23:0	Reserved.

MSR0000_0408 MC2 Machine Check Control (MC2_CTL)

Read-write; Per-compute-unit.
Reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSRC001_0046 [BU Machine Check Control Mask (MC2_CTL_MASK)].

Bits	Description
63:16	Unused.
15	**HWA: Hardware assertion**.
14	**L2TlbPoison: TLB fill poison error from L2**.
13	**RdData: read data error from NB**.
12	**L2Tag: L2 cache tag error**.
11	**L2TlbData: L2 TLB parity error**. Parity error reading from TLB.
10	**L2Prefetch: L2 data prefetcher parity error**.
9	**XabAddr: XAB address parity error**.
8	**PrbAddr: probe buffer address parity error**.
7	**FillData: fill data parity and ECC error**.
6	**PrqAddr: post retire queue address parity error**.
5	**PrqData: post retire queue data parity error**.
4	**WccAddr: write coalescing cache address ECC error**.
3	**WccData: write coalescing cache data ECC error**.
2	**WcbData: write combining buffer data parity error**.
1	**VbData: victim buffer data parity and ECC error**.
0	**L2TagMultiHit: L2 tag multiple hit error**.

MSR0000_0409 MC2 Machine Check Status (MC2_STATUS)

See 2.16.1 [Machine Check Architecture]. See MSRC001_0015[McStatusWrEn]. Table 246 describes each error type. Table 247 describes the error codes and status register settings for each error type.

Bits	Description
63	**Val: error valid**. See: MSR0000_0401[Val].
62	**Overflow: error overflow**. See: MSR0000_0401[Overflow].
61	**UC: error uncorrected**. See: MSR0000_0401[UC].
60	**En: error enable**. See: MSR0000_0401[En].
59	**MiscV: miscellaneous error register valid**. Read-write; Updated-by-hardware. Cold reset: 0. See: MSR0000_0401[MiscV]. 1=Valid thresholding in MSR0000_040B.
58	**AddrV: error address valid**. See: MSR0000_0401[AddrV].
57	**PCC: processor context corrupt**. See: MSR0000_0401[PCC].
56:55	Reserved.
54:47	**Syndrome[7:0]**. Read-write; Updated-by-hardware. Cold reset: 0. The syndrome bits when an ECC error is detected. See Table 247 for when Syndrome[11:0] is valid. Syndrome[11:0] = {Syndrome[11:8], Syndrome[7:0]}. Array — Description L2 Tag — Syndrome[7:0]. WCC Tag — Syndrome[11:0]. L2 Data — Syndrome[8:0]. WCC Data — Syndrome[8:0].
46	**CECC: correctable ECC error**. Read-write; Updated-by-hardware. Cold reset: 0. 1=The error was a correctable ECC error.
45	**UECC: uncorrectable ECC error**. Read-write; Updated-by-hardware. Cold reset: 0. 1=The error was an uncorrectable ECC error.
44	**Deferred: deferred error**. See: MSR0000_0401[Deferred].
43	**Poison: poison error**. See: MSR0000_0401[Poison].
42:40	Reserved.
39:36	**Way: cache way in error**. Read-write; Updated-by-hardware. Cold reset: 0. Indicates the cache way in error. See Table 247 for when Way is valid and what ways are valid. Bits — Description 0h — Way 0 1h — Way 1 Eh-2h — Way <Way> Fh — Way 15
35:32	Reserved.
31:28	Reserved.
27:24	**Syndrome[11:8]**. See: MSR0000_0409[Syndrome[7:0]].
23:22	Reserved.

21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware. Cold reset: 0. See MSR0000_0401[ErrorCodeExt]. See Table 247 for expected values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware. Cold reset: 0. See 2.16.1.5 [Error Code] for details on decoding this field. See Table 247 for expected values.

Table 245: MBE, SBU, and SBC Definitions

Term	Definition
MBE	Multi-bit ECC error, uncorrected.
SBU	Single-bit ECC error, not-corrected. There are some implementation specific conditions when a single bit error is not correctable.
SBC	Single-bit ECC error is detected and correctable.

Table 246: MC2 Error Descriptions

Error Type	Error	Description	CTL[3]	CID[2]	EAC[1]
System Read Data	L2Tlb Prefetch Wcc	An error occurred during an attempted read of data from the NB. Possible reasons include master abort, target abort, and receipt of read data error for TLB. Error Action[4]: None.	RdData	A	D
TLB	TlbPar	Data parity error reading from TLB. Error Action[4]: Invalidate TLB entry.	L2TlbData	A	D
	FillErr	Poison data provided for TLB fill. Error Action[4]: None.	L2TlbPoison	A	D
L2 Cache	Prefetch	Prefetcher request FIFO parity error. Error Action[4]: Invalidate entry (drop prefetch).	L2Prefetch	A	D
L2 Cache	FillEcc	Fill ECC error on data fills. CECC: Corrected data returned to destination; error remains in source. UECC: Poison data returned to destination; error remains in source. The data sources are indicated in LL field and affect what part of Way is valid: Source LL Way WCC L1 [1:0]. See Note 1. L2 L2 [3:0]. NB LG -. See Note 2. Notes: 1. WCC: Indicates data corrupted in WCC or Fill Buffer. 2. NB: Note: Data from NB was sent either okay (good ECC) or already poisoned. Indicates data corrupted in Fill Buffer.	FillData	A	D

Table 246: MC2 Error Descriptions (Continued)

Error Type	Error	Description	CTL[3]	CID[2]	EAC[1]
L2 Cache	FillPar	Fill parity error on instruction fills. SubCase — Description NB->IC — NB to IC parity error: Error Action[4]: Invalidate data and Nack request (IC will re-request). L2->IC — L2 to IC parity error: Error Action[4]: Invalidate data and Nack request (IC will re-request. L2->LS,TLB — L2 to LS or TLB parity error: Error Action[4]: Poison data returned to destination; error remains in source.	FillData	A	D
L2 Cache	PrqAddr	Post Retire Queue address parity error. Error Action[4]: Sync flood.	PrqAddr	A	D
L2 Cache	PrqData	Post Retire Queue data parity error. Error Action[4]: Poison line WCC or line sent to NB.	PrqData	A	D
L2 Cache	WccTag	Write Coalescing Cache tag ECC error. SubCase — Error Action[4] UECC — Sync flood. CECC — Invalidate Wcc tag entry (cleans error).	WccAddr	0	D
L2 Cache	WccData	WCC data ECC error. SubCase — Error Action[4] UECC — Poison copy in WCC. CECC — Corrected copy in WCC.	WccData	A	D
L2 Cache	WcbData	WCB data parity error.Error Action[4]: Poison sent to NB.	WcbData	A	D
L2 Cache	VbData	VB data ECC or parity error. SubCase — Description Par — Parity: Parity error indicated when CECC and UECC are both clear. Error Action[4]: Poison sent to NB. UECC — Error Action[4]: Poison sent to NB. CECC — Single-bit ECC error, corrected: Error Action[4]: Corrected data sent to NB.	VbData	0	D
L2 Cache	L2TagMH	Multiple hits on L2 tag. Error Action[4]: Sync flood.	L2TagMultiHit	0	D

Table 246: MC2 Error Descriptions (Continued)

Error Type	Error	Description	CTL[3]	CID[2]	EAC[1]
L2 Cache	L2Tag	A correctable or uncorrectable ECC error was seen in the L2 tag. The L2TagMH error signature supersedes the L2Tag error signature if they both occur for the same L2 tag read. SubCase — Description UECC — Error Action[4]: Sync flood. CECC — Error Action[4]: Correct error in array and retry the operation. Hard — A hard correctable error (UC, CECC) was seen in the L2 tag. Error Action[4]: Sync flood.	L2Tag	0	D
L2 Cache	XabAddr	Transaction Address Buffer (XAB) parity error. This error is system fatal; memory coherence may have been affected. Error Action[4]: Sync flood.	XabAddr	A	D
L2 Cache	PrbAddr	Probe buffer address parity error. This error is system fatal; memory coherence may have been affected. Error Action[4]: Sync flood.	PrbAddr	0	D
Internal Error	Hardware Assertion	Ahardware assertion was detected.	HWA	0	D

1. EAC: The error action is taken if detected for all CU errors. D=Error action taken if detected. E=Error action taken if MCA bank enabled. See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
2. CID: core ID. A=Error reported to the affected core. 0=Error reported to core 0 of the compute unit; see 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
3. See MSR0000_0408.
4. Error Action: Sync flood=Take sync flood if PCC=1. None=No action other than that specified by MCA.

Table 247: MC2 Error Signatures

Error Type	Error Sub-Type	Sub	ErrorCodeExt	Error Code						UC	ADDRV	PCC	Syndrome	Way	CECC	UECC	Deferred	Poison
				Type	UU/PP	T	RRRR	II/TT	LL									
TLB	TlbPar	-	00h	TLB	-	-	-	G	L2	0	1	0	-	[2:0]	0	0	0	0
	FillErr	-	01h							1				-				1
System Read Data	TLB	-	00h	BUS	SRC	0	RD	MEM/IO	L2	1	1	0	-	-	0	0	0	0
	Prefetch		01h					MEM	L2	0								
	Wcc		02h				DWR		L1	1								
L2 Cache	FillEcc	-	04h	MEM	-	-	DRD	D	See[1]	0	1	0	[8:0]	See[1]	0/1	0/1	0/1	0
L2 Cache	FillPar	NB->IC	05h	MEM	-	-	IRD	I	LG	0	0	0	-	-	0	0	0	0
		L2->IC							L2									
		L2->					DRD	D		0	1	0		[3:0]			1	

Table 247: MC2 Error Signatures (Continued)

Error Type	Error Sub-Type	Sub	ErrorCodeExt	Error Code						UC	ADDRV	PCC	Syndrome	Way	CECC	UECC	Deferred	Poison
				Type	UU/PP	T	RRRR	II/TT	LL									
L2 Cache	Prefetch	-	06h	MEM	-	-	Prefetch	D	L2	0	1	0	-	-	0	0	0	0
	PrqAddr	-	07h				DWR	D	L1	1	0	1	-	-	0	0	0	0
	PrqData	-	08h							0		0					1	
L2 Cache	WccTag	-	09h	MEM	-	-	DWR	D	L1	0/1	1	UC	[11:0]	[1:0]	0/1	0/1	0	0
	WccData	-	0Ah						L1	0	1	0	[8:0]	[1:0]	0/1	0/1	0/1	0
	WcbData	-	0Bh						LG	0		0	-	-	0	0	1	
L2 Cache	VbData	Par	0Ch	MEM	-	-	Probe, Evict	I	L2	0	0	0	-	-	0	0	1	0
		ECC						D					[8:0]		0/1	0/1		
Tag	L2Tag	-	10h	MEM	-	-	GEN	G	L2	0/1	1	UC	[7:0]	[3:0]	0/1	0/1	0	0
		Hard	11h							1	1	1	-	[3:0]	0	0	0	0
	L2Tag	-	12h							1	1	1	-	-	0	0	0	0
	XabAddr	-	13h															
	PrbAddr	-	14h				Probe											
Internal Error	HWA	-	00h-3Fh	INT	HWA	-	-	-	-	1	0	1	-	-	0	0	0	0

1. LL and Way are specified in Table 246.

MSR0000_040A MC2 Machine Check Address (MC2_ADDR)

Read-write; Updated-by-hardware. Cold reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. The following table defines the address register as a function of error type.

Bits	Description
63:0	**ADDR**. Read-write; Updated-by-hardware. See Table 248.

Table 248: MC2 Address Register

Error Type	Error Sub-Type	Bits	Description
System Read Data Error	-	63:48	Reserved
		47:6	**PhysAddr[47:6]**.
		5:0	Reserved
TLB	TlbPar	63:7	Reserved
		6:0	**Index[6:0]**.
	FillErr	63:48	Reserved
		47:6	**PhysAddr[47:6]**.
		5:0	Reserved

Table 248: MC2 Address Register (Continued)

Error Type	Error Sub-Type	Bits	Description
L2 Cache	Prefetch	63:5	Reserved
		4:0	Prefetch FIFO read pointer.
L2 Cache	FillEcc, FillPar (SubCase=L2->LS,TLB)	63:48	Reserved
		47:3	**PhysAddr[47:3]**.
		2:0	Reserved
L2 Cache	WccTag, WccData	63:10	Reserved
		9:6	**Index[9:6]**.
		5:0	Reserved
L2 Cache	WcbData	63:48	Reserved
		47:3	**PhysAddr[47:3]**.
		2	Reserved
		1:0	Index[1:0].
Tag	L2Tag, L2TagMH	63:17	Reserved
		16:6	**PhysAddr[16:6]**. Bits Description 1 MB PhysAddr[15:6] valid; [16] Reserved. 2 MB PhysAddr[16:6] valid See CPUID Fn8000_0006_ECX[L2Size]
		5:0	Reserved
Tag	XabAddr	63:5	Reserved
		4:0	XabIndex.
Tag	PrbAddr	63:4	Reserved
		3:0	ProbeBufferIndex.

MSR0000_040B MC2 Machine Check Miscellaneous (MC2_MISC)

Cold reset: 0000_0000_0000_0000h.
See 2.16.1.7 [Error Thresholding].

Bits	Description
63	**Valid**. See: MSR0000_0403[Valid].
62	**CntP: counter present**. See: MSR0000_0403[CntP].
61	**Locked**. See: MSR0000_0403[Locked].
60	**IntP: Interrupt support present.** See: MSR0000_0403[IntP].
59:56	Reserved.
55:52	**LvtOffset: LVT offset**. See: MSR0000_0403[LvtOffset].
51	**CntEn: counter enable**. See: MSR0000_0403[CntEn].
50:49	**IntType: interrupt type**. See: MSR0000_0403[IntType].
48	**Ovrflw: overflow.** See: MSR0000_0403[Ovrflw].

47:44	Reserved.
43:32	**ErrCnt: error counter**. See: MSR0000_0403[ErrCnt].
31:24	**BlkPtr: Block pointer for additional MISC registers**. See: MSR0000_0403[BlkPtr].
23:0	Reserved.

MSR0000_040C MC3 Machine Check Control (MC3_CTL)

Reset: 0000_0000_0000_0000h. Read-only.

Bits	Description
63:0	Unused.

MSR0000_040D MC3 Machine Check Status (MC3_STATUS)

Reset: 0. See MSRC001_0015[McStatusWrEn].

Bits	Description
63:0	Reserved.

MSR0000_040E MC3 Machine Check Address (MC3_ADDR)

Reset: 0000_0000_0000_0000h. Read-only.

Bits	Description
63:0	Reserved.

MSR0000_040F MC3 Machine Check Miscellaneous (MC3_MISC)

Reset: 0000_0000_0000_0000h. Read-only.

Bits	Description
63:0	Reserved.

MSR0000_0410 MC4 Machine Check Control (MC4_CTL)

Read-write; Not-same-for-all. Reset: 0000_0000_0000_0000h.
MSR0000_0410[31:0] is an alias of D18F3x40, which is accessible through PCI configuration space. Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn].

See D18F3x44 [MCA NB Configuration] for further NB MCA configuration controls. See 2.16.1 [Machine Check Architecture] for a general description of the machine check architecture. See MSRC001_0048 [NB Machine Check Control Mask (MC4_CTL_MASK)] for the corresponding error mask register.

Bits	Description
63:32	Unused.
31:0	Alias of D18F3x40.

MSR0000_0411 MC4 Machine Check Status (MC4_STATUS)

Not-same-for-all. Cold reset: 0000_0000_0000_0000h.
MSR0000_0411[63:32] is an alias of D18F3x4C, and MSR0000_0411[31:0] is an alias of D18F3x48. Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. See MSRC001_0015[McStatusWrEn] for information on writing to this register. See 2.16.1 [Machine Check Architecture] for machine check architecture background.

Table 249 describes each error type. Table 250 and Table 251 describe the error codes and status register settings for each error type.

Bits	Description
63	**Val: valid**. See: MSR0000_0401[Val].
62	**Overflow: error overflow**. See: MSR0000_0401[Overflow].
61	**UC: error uncorrected**. See: MSR0000_0401[UC].
60	**En: error enable**. See: MSR0000_0401[En].
59	**MiscV: miscellaneous error register valid**. Read-write; Updated-by-hardware. 1=Valid thresholding in MSR0000_0413 or MSRC000_0408.
58	**AddrV: error address valid**. See: MSR0000_0401[AddrV].
57	**PCC: processor context corrupt**. See: MSR0000_0401[PCC].
56	**ErrCoreIdVal: error core ID is valid**. Read-write; Set-by-hardware. 1=The ErrCoreId field is valid.
55:45	Reserved.
44	**Deferred: deferred error**. Read-write; Updated-by-hardware. 1=A deferred error was created. A deferred error is the result of an uncorrectable data error which did not immediately cause a processor exception; the data is poisoned and an exception is deferred until the data is consumed.
43:42	Reserved.
41	**SubLink: sublink**. Read-write; Set-by-hardware. For errors associated with a link, this bit indicates if the error was associated with the upper or lower byte of the link. 0=Sublink [7:0]. 1=Sublink [15:8].
40:36	Reserved.
35:32	**ErrCoreId: error associated with core N**. Read-write; Updated-by-hardware. When ErrCoreIdVal == 1 this field indicates which core within the processor is associated with the error; otherwise this field is reserved. All values greater than D18F5x84[CmpCap] are reserved.
31:22	Reserved.
21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware. See MSR0000_0401[ErrorCodeExt]. See Table 250 for values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware. See 2.16.1.5 [Error Code].

Table 249: MC4 Error Descriptions

Error Type	Description	CTL[1]	ETG[2]	EAC[4]
Sync Error	Link-defined sync error packets detected on link. The NB floods its outgoing links with sync packets after detecting a sync packet on an incoming link independent of the state of the control bits.	SyncPktEn	L	D
Master Abort	Master abort seen as result of link operation. Reasons for this error include requests to non-existent addresses. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of the control bit.	MstrAbortEn	L	D
Target Abort	Target abort seen as result of link operation. The NB returns an error response back to the requestor with any associated data all 1s independent of the state of the control bit.	TgtAbortEn	L	D
RMW Error	An atomic read-modify-write (RMW) command was received from an IO link. Atomic RMW commands are not supported. An atomic RMW command results in a link error response being generated back to the requesting IO device. The generation of the link error response is not affected by the control bit.	AtomicRMWEn	L	D
WDT Error	NB WDT timeout due to lack of progress. The NB WDT monitors transaction completions. A transaction that exceeds the programmed time limit reports errors via the MCA. The cause of error may be another node or device which failed to respond.	WDTRptEn	L	D
DRAM ECC Error	A DRAM ECC error detected.	CECCEn, UECCEn	D	D
Link Data Error	Data error detected on link. If enabled for reporting and the request is sourced from a core, then PCC is set. (If not enabled for reporting, PCC is not set. If configured to allow an error response to be returned to the core, this could allow error containment to a scope smaller than the entire system.)	McaUsPwDatErrEn, CpPktDatEn	L	D
Protocol Error	Protocol error detected by link. These errors are distinguished from each other by the value in MSR0000_0412[ErrAddr]. See Table 186. For protocol errors, the system cannot continue operation. Protocol errors can be caused by other subcomponents than the one reporting the error. For diagnosis, collect and examine MCA registers from other banks, cores, and processors in the system.	NbIntProtEn	L[3]	D
NB Array Error	A parity error was detected in the NB internal arrays.	NbArrayParEn	-	D

Table 249: MC4 Error Descriptions (Continued)

Error Type	Description	CTL[1]	ETG[2]	EAC[4]
Compute Unit Data Error	NB received a data error from a core and this error could not be contained. For the cause of the data error, examine the core MCA registers for deferred errors. This error may occur for the following types of data writes: • APIC • Configuration space (IO and MMIO) For these errors, sync flood will occur if D18F3x180[SyncFloodOnCpuLeakErr] is set.	McaCpuDatErrEn	-	D

1. CTL: See MSR0000_0410.
2. ETG: error threshold group. See 2.16.1.7 [Error Thresholding].
 - L=Link.
 - D=DRAM.
3. The error thresholding group is Link if link protocol error; none for non-link protocol error.
4. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled.

The NB is capable of reporting the following errors:

Table 250: MC4 Error Signatures, Part 1

Error Type	ErrorCode-Ext	Error Code					
		Type	PP	T	RRRR	II/TT	LL
Reserved	00h	-	-	-	-	-	-
Reserved	01h	-	-	-	-	-	-
Sync Error	02h	BUS	OBS	0	GEN	GEN	LG
Mst Abort	03h	BUS	SRC/OBS	0	RD/WR	MEM/IO[1]	LG
Tgt Abort	04h	BUS	SRC/OBS	0	RD/WR	MEM/IO[1]	LG
RMW Error	06h	BUS	OBS	0	GEN	IO	LG
WDT Error	07h	BUS	GEN	1	GEN	GEN	LG
Link Data Error	0Ah	BUS	SRC/OBS	0	RD/WR/ DWR	MEM/IO	LG
NB Protocol Error	0Bh		OBS	0	GEN	GEN	LG
NB Array Error	0Ch		OBS	0	GEN	GEN	LG
Compute Unit Data Error	19h	MEM	-	-	WR	Data	LG

1. Indicates the type of link attached to the reporting NB, not the instruction type. MEM indicates coherent link, IO indicates IO link.

Table 251: MC4 Error Signatures, Part 2

Error Type	UC	AddrV	PCC	Syndrome Valid	CECC	UECC	Deferred	Scrub	Link	Err CoreId
Sync Error	1	0	1	-	0	0	0	0	Y	-
Mst Abort	1	1	Core[10]	-	0	0	0	0	Y	Y
Tgt Abort	1	1	Core[10]	-	0	0	0	0	Y	Y
RMW Error	1	1	0	-	0	0	0	0	Y	-

Table 251: MC4 Error Signatures, Part 2 (Continued)

Error Type	UC	AddrV	PCC	Syndrome Valid	CECC	UECC	Deferred	Scrub	Link	Err CoreId
WDT Error	1	0[1] 1[2]	1	-	0	0	0	0	-	-
ECC Error	0/1	1	0/1	15:0	0/1[13]	0/1[13]	0	1/0	-	-
ECC Error, Deferred	0	1	0	15:0	0/1[13]	0/1[13]	1	1/0	-	-
Link Data Error	~Deferred[11]	1	0	-	0	0	0/1	0	Y	-
NB Protocol Error	1	1/0[2]	1	-	0	0	0	0	Y	-
NB Array Error	~Deferred[11]	1[4]	~Deferred[11]	-	0	0	0/1	0	-	-
Compute Unit Data Error	1	0	1	-	0	0	0	0	-	Y

1. See Table 190 [Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Watchdog Timer Errors].
2. See Table 186 [Format of {D18F3x54[ErrAddr[47:32]], D18F3x50[ErrAddr[31:1]]} for Protocol Errors].
4. See Table 189 [Valid Values for ArrayErrorType].
10. Core: source is core. 1=Source is core. 0=Source is not core.
13. Hist: Error was detected by the hardware-managed history scheme.

MSR0000_0412 MC4 Machine Check Address (MC4_ADDR)

IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSE Read-write; Per-node; Not-same-for-all. ENDIF. Cold reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. MSR0000_0412[31:0] is an alias of D18F3x50. MSR0000_0412[63:32] is an alias of D18F3x54. Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. ErrAddr[47:1] carries supplemental information associated with a machine check error, generally the address being accessed. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. The format of ErrAddr[47:1] is a function of MSR0000_0411[ErrorCodeExt]; See ErrAddr[47:1].

Bits	Description
63:48	Reserved. Value: 00h.
47:32	**ErrAddr[47:32]: Error Address Bits[47:32]**. See: ErrAddr[31:1].
31:1	**ErrAddr[31:1]: Error Address Bits[31:1]**. ErrAddr[47:0] = {MSR0000_0412[ErrAddr[47:32]], MSR0000_0412[ErrAddr[31:1]], MSR0000_0412[ErrAddr[0]]}. See the tables Table 185 - Table 190 for the encoding.
0	**ErrAddr[0]: Error Address Bit[0]**.

The register format depends on the type of error being logged:

- Protocol errors contain the error reason code, may contain the physical address, and are formatted according to Table 186.
- NB array errors indicate the array in error, and are formatted according to Table 188.
- NB Watchdog timer errors depend on the mode selected by D18F3x180[McaLogErrAddrWdtErr], and the format is indicated by D18F3x4C[AddrV]. If D18F3x4C[AddrV] is indicated, errors are formatted according to Table 185. If D18F3x4C[AddrV] is not indicated, errors are formatted according to Table 190.
- All other NB errors which indicate D18F3x4C[AddrV] are formatted according to Table 185.

MSR0000_0413 NB Machine Check Misc 4 (MC4_MISC0)

Bits	Description
63:32	Reserved.
31:24	**BlkPtr: Block pointer for additional MISC registers**. Read-only. Value: 01h. 01h=Extended MC4_MISC MSR block is valid. See MSRC000_0408.
23:0	Reserved.

MSR0000_0414 MC5 Machine Check Control (MC5_CTL)

Read-write. Reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSRC001_0049 [EX Machine Check Control Mask (MC5_CTL_MASK)].

Bits	Description
63:16	Unused.
15	Unused. Read-write.
14	**HWA: hardware assertion**.
13	Unused. Read-write.
12	**DE: DE error**.
11	**FRF: flag register file parity**.
10	**AG1PRF: physical register file AG1 port parity**.
9	**AG0PRF: physical register file AG0 port parity**.
8	**EX1PRF: physical register file EX1 port parity**.
7	**EX0PRF: physical register file EX0 port parity**.
6	**MAP: mapper checkpoint array parity**.
5	**RETDISP: retire dispatch queue parity**.
4	**IDF: IDRF array parity**.
3	**PLDEX: EX payload array parity**.
2	**PLDAG: AG payload array parity**.
1	Reserved.
0	**WDT: core watchdog timer**. See MSRC001_0074 [CPU Watchdog Timer (CpuWdtCfg)].

MSR0000_0415 MC5 Machine Check Status (MC5_STATUS)

Cold reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSRC001_0015[McStatusWrEn]. Table 252 describes each error type. Table 253 describes the error codes and status register settings for each error type.

Bits	Description
63	**Val: error valid**. See: MSR0000_0401[Val].
62	**Overflow: error overflow**. See: MSR0000_0401[Overflow].
61	**UC: error uncorrected**. See: MSR0000_0401[UC].
60	**En: error enable**. See: MSR0000_0401[En].
59	**MiscV: miscellaneous error register valid**. Read-write; Updated-by-hardware. See: MSR0000_0401[MiscV]. 1=Valid thresholding in MSR0000_0417.
58	**AddrV: error address valid**. See: MSR0000_0401[AddrV]. 1=Valid address in MSR0000_0416.
57	**PCC: processor context corrupt**. See: MSR0000_0401[PCC].
56:22	Reserved.
21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware. See MSR0000_0401[ErrorCodeExt]. See Table 253 for values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware. See 2.16.1.5 [Error Code] See Table 253.

Table 252: MC5 Error Descriptions

Error Type	Error Sub-type	Description	CTL[1]	EAC[3]
WDT error	-	The WDT timer has expired. See MSRC001_0074 [CPU Watchdog Timer (CpuWdtCfg)].	WDT	E

Table 252: MC5 Error Descriptions (Continued)

Error Type	Error Sub-type	Description	CTL[1]	EAC[3]
Internal	AG payload array parity	A parity error occurred in the address generator payload array.	PLDAG	D
	EX payload array parity	A parity error occurred in the EX payload array.	PLDEX	D
	IDRF array parity	A parity error occurred in the immediate displacement register file.	IDF	D
	Retire dispatch queue parity	A parity error occured in the retire dispatch queue. This error causes the processor to enter the Shutdown state; 2.16.1.3.1 [MCA Conditions that Cause Shutdown].	RETDISP	E
	Mapper checkpoint array parity	A parity error occured in the mapper checkpoint array. This error causes the processor to enter the Shutdown state if UC=1; 2.16.1.3.1 [MCA Conditions that Cause Shutdown].	MAP	D
	EX0PRF parity	A parity error occurred in the physical register file's EX0 port.	EX0PRF	D
	EX1PRF parity	A parity error occurred in the physical register file's EX1 port.	EX1PRF	D
	AG0PRF parity	A parity error occurred in the physical register file's AG0 port.	AG0PRF	D
	AG1PRF parity	A parity error occurred in the physical register file's AG1 port.	AG1PRF	D
	Flag register file parity	A parity error occurred in the flag register file.	FRF	D
	DE error	A DE error occurred.	DE	E
Internal	Hardware Assertion	Ahardware assertion was detected.	HWA	D

1. CTL: See MSR0000_0414.
2. CID: core ID. All EX errors are reported to the affected core; see 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
3. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled. See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].

Table 253: MC5 Error Signatures

Error Type	Error Sub-Type	ErrorCodeExt	Error Code						UC	ADDRV	MISCV	PCC	CECC	UECC
			Type	UU/PP	TT	RRRR	II/TT	LL						
WDT error	-	00h	BUS	GEN	1	GEN	GEN	LG	1	1	1	1	0	0

Table 253: MC5 Error Signatures (Continued)

Error Type	Error Sub-Type	ErrorCodeExt	Error Code						UC	ADDRV	MISCV	PCC	CECC	UECC
			Type	UU/PP	TT	RRRR	II/TT	LL						
Internal error	AG payload array parity	02h	BUS	GEN	0	GEN	GEN	LG	1/0	1	0	0	0	0
	EX payload array parity	03h							1/0	1	0	0	0	0
	IDRF array parity	04h							1/0	1	0	0	0	0
	Retire dispatch queue parity	05h							1[1]	1	0	0	0	0
	Mapper checkpoint array parity	06h							1/0[2]	1	0	0	0	0
	EX0PRF parity	07h							1/0	0	0	0	0	0
	EX1PRF parity	08h							1/0	0	0	0	0	0
	AG0PRF parity	09h							1/0	0	0	0	0	0
	AG1PRF parity	0Ah							1/0	0	0	0	0	0
	Flag register file parity	0Bh							1/0	0	0	0	0	0
	DE error	0Ch							1/0[2]	0	0	0	0	0
Internal error	HWA	00h-1Fh	INT	HWA	0	-	-	-	1	0	0	1	0	0

1. Causes shutdown.
2. Causes shutdown if UC=1.
3. STATQ sets PCC=1 if a store retired in last 3 cycles or FpTokenRet bits mismatch.

MSR0000_0416 MC5 Machine Check Address (MC5_ADDR)

Read-write; Updated-by-hardware. Cold reset: 0000_0000_0000_0000h. The MCi_ADDR register contains valid data if indicated by MCi_STATUS[AddrV]. See 2.16.1 [Machine Check Architecture]. The register format depends on the type of error being logged.

Bits	Description
63:0	**ADDR**. See Table 254.

The following tables define the address register as a function of error type.

Table 254: MC5 Address Register

Error Type	Error Sub-Type	Bits	Description
WDT	- (ErrorCodeExt=00h)	63:48	Reserved
		47:0	**LogAddr[47:0]**. Logical address of the next instruction after the last instruction retired.

Table 254: MC5 Address Register (Continued)

Error Type	Error Sub-Type	Bits	Description
Internal	AG payload array parity (ErrorCodeExt=02h), EX payload array parity (ErrorCodeExt=03h), IDRF array parity (ErrorCodeExt=04h)	63:6	Reserved
		5:0	**SchedulerQID**.
Internal	Retire dispatch queue parity (ErrorCodeExt=05h)	63:7	Reserved
		6:0	**RetirementID**.
Internal	Mapper checkpoint array parity (ErrorCodeExt=06h)	63:6	Reserved
		5:0	**CheckpointID**.

MSR0000_0417 MC5 Machine Check Miscellaneous (MC5_MISC)

Cold reset: 0000_0000_0000_0000h. This register records unspecified, implementation-specific status bits when an FR machine check error is logged.
See 2.16.1.7 [Error Thresholding].

Bits	Description
63	**Valid**. See: MSR0000_0403[Valid].
62	**CntP: counter present**. See: MSR0000_0403[CntP].
61	**Locked**. See: MSR0000_0403[Locked].
60	**IntP: Interrupt support present.** See: MSR0000_0403[IntP].
59:56	Reserved.
55:52	**LvtOffset: LVT offset**. See: MSR0000_0403[LvtOffset].
51	**CntEn: counter enable**. See: MSR0000_0403[CntEn].
50:49	**IntType: interrupt type**. See: MSR0000_0403[IntType].
48	**Ovrflw: overflow.** See: MSR0000_0403[Ovrflw].
47:44	Reserved.
43:32	**ErrCnt: error counter**. See: MSR0000_0403[ErrCnt].
31:24	**BlkPtr: Block pointer for additional MISC registers**. See: MSR0000_0403[BlkPtr].
23:0	Reserved.

MSR0000_0418 MC6 Machine Check Control (MC6_CTL)

Per-compute-unit; Read-write. Reset: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSRC001_004A [FP Machine Check Control Mask (MC6_CTL_MASK)].

Bits	Description
63:7	Unused.
6	**HWA: Hardware assertion**.
5	**SRF: status register file parity error**.
4	**RetireQ: retire queue parity error**.

3	Unused.
2	**Sched: scheduler table parity error**.
1	**FreeList: free list parity error**.
0	**PRF: physical register file parity error**.

MSR0000_0419 MC6 Machine Check Status (MC6_STATUS)

See 2.16.1 [Machine Check Architecture]. See MSRC001_0015[McStatusWrEn]. Table 255 describes each error type. Table 256 describes the error codes and status register settings for each error type.

Bits	Description
63	**Val: valid**. See: MSR0000_0401[Val].
62	**Overflow: error overflow**. See: MSR0000_0401[Overflow].
61	**UC: error uncorrected**. See: MSR0000_0401[UC].
60	**En: error enable**. See: MSR0000_0401[En].
59	**MiscV: miscellaneous error register valid**. Read-only. Value: 0. See: MSR0000_0401[MiscV].
58	**AddrV: error address valid**. Read-only. Value: 0. See: MSR0000_0401[AddrV].
57	**PCC: processor context corrupt**. See: MSR0000_0401[PCC].
56:22	Reserved.
21:16	**ErrorCodeExt: extended error code**. Read-write; Updated-by-hardware. Logs an extended error code when an error is detected. This model-specific field is used in conjunction with ErrorCode to identify the error sub-type for root cause analysis (see 2.16.1.5 [Error Code]). See Table 256 for expected values.
15:0	**ErrorCode: error code**. Read-write; Updated-by-hardware. See 2.16.1.5 [Error Code] for details on decoding this field. See Table 256 for expected values.

Table 255: MC6 Error Descriptions

Error Type	Error	Description[2]	CTL[4]	CID[3]	EAC[1]
Floating Point Unit	Physical Register File	A parity error occurred in the Physical Register File (PRF).	PRF	0	E
	Status Register File	A parity error occurred in the Status Register File (SRF).	SRF	A	E
	Free List	A parity error occurred on the Free List.	FreeList	0	E
	Retire Queue	A parity error occurred in the Retire Queue.	RetireQ	0	E
	Scheduler	A parity error occurred in the Scheduler table.	Sched	0	E
Internal Error	Hardware Assertion	Ahardware assertion was detected.	HWA	0	D

Table 255: MC6 Error Descriptions (Continued)

Error Type	Error	Description[2]	CTL[4]	CID[3]	EAC[1]

1. EAC: D=Error action taken if detected. E=Error action taken if MCA bank enabled. See 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
2. All FP errors are system fatal and result in a Sync flood.
3. CID: core ID. A=Error reported to the affected core. 0=Error reported to core 0 of the compute unit; see 2.16.1.3 [Error Detection, Action, Logging, and Reporting].
4. See MSR0000_0418.

Table 256: MC6 Error Signatures

Error Type	Error Sub-Type	ErrorCodeExt	Error Code						UC	ADDRV	PCC
			Type	UU/PP	T	RRRR	II	LL			
Floating Point Unit	Status Register File	00101b	BUS	GEN	0	GEN	GEN	LG	1	0	1
	Physical Register File	00010b									
	Free List	00001b									
	Retire Queue	00011b									
	Scheduler	00100b									
Internal Error	Hardware Assertion	00000b	INT	HWA	0	-	-	-	1	0	1

MSR0000_041A MC6 Machine Check Address (MC6_ADDR)

Reset: 0000_0000_0000_0000h. Read-only. See 2.16.1 [Machine Check Architecture].

Bits	Description
63:0	Reserved.

MSR0000_041B MC6 Machine Check Miscellaneous (MC6_MISC)

Reset: 0000_0000_0000_0000h. Read-only. See 2.16.1 [Machine Check Architecture].

Bits	Description
63:0	Reserved.

3.21 MSRs - MSRC000_0xxx

MSRC000_0080 Extended Feature Enable (EFER)

SKINIT Execution: 0000_0000_0000_0000h.

Bits	Description
63:16	MBZ.
15	**TCE: translation cache extension enable**. Read-write. Reset: 0. 1=Translation cache extension is enabled.
14	**FFXSE: fast FXSAVE/FRSTOR enable**. Read-write. Reset: 0. 1=Enables the fast FXSAVE/FRSTOR mechanism. A 64-bit operating system may enable the fast FXSAVE/FRSTOR mechanism if (CPUID Fn8000_0001_EDX[FFXSR] == 1)). This bit is set once by the operating system and its value is not changed afterwards.
13	**LMSLE: long mode segment limit enable**. Read-write. Reset: 0. 1=Enables the long mode segment limit check mechanism.
12	**SVME: secure virtual machine (SVM) enable**. IF (MSRC001_0114[SvmeDisable] == 1) THEN MBZ. ELSE Read-write. ENDIF. Reset: 0. 1=SVM features are enabled.
11	**NXE: no-execute page enable**. Read-write. Reset: 0. 1=The no-execute page protection feature is enabled.
10	**LMA: long mode active**. Read-only. Reset: 0. 1=Indicates that long mode is active.
9	MBZ.
8	**LME: long mode enable**. Read-write. Reset: 0. 1=Long mode is enabled.
7:1	RAZ.
0	**SYSCALL: system call extension enable**. Read-write. Reset: 0. 1=SYSCALL and SYSRET instructions are enabled. This adds the SYSCALL and SYSRET instructions which can be used in flat addressed operating systems as low latency system calls and returns.

MSRC000_0081 SYSCALL Target Address (STAR)

Reset: 0000_0000_0000_0000h. This register holds the target address used by the SYSCALL instruction and the code and stack segment selector bases used by the SYSCALL and SYSRET instructions.

Bits	Description
63:48	**SysRetSel: SYSRET CS and SS**. Read-write.
47:32	**SysCallSel: SYSCALL CS and SS**. Read-write.
31:0	**Target: SYSCALL target address**. Read-write.

MSRC000_0082 Long Mode SYSCALL Target Address (STAR64)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**LSTAR: long mode target address**. Read-write. Target address for 64-bit mode calling programs. The address stored in this register must be in canonical form (if not canonical, a #GP fault occurs).

MSRC000_0083 Compatibility Mode SYSCALL Target Address (STARCOMPAT)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**CSTAR: compatibility mode target address**. Read-write. Target address for compatibility mode. The address stored in this register must be in canonical form (if not canonical, a #GP fault occurs).

MSRC000_0084 SYSCALL Flag Mask (SYSCALL_FLAG_MASK)

Bits	Description
63:32	RAZ.
31:0	**Mask: SYSCALL flag mask**. Read-write. Reset: 0000_0000h. This register holds the EFLAGS mask used by the SYSCALL instruction. 1=Clear the corresponding EFLAGS bit when executing the SYSCALL instruction.

MSRC000_00E7 Read-Only Max Performance Frequency Clock Count (MPerfReadOnly)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**MPerfReadOnly: read-only maximum core clocks counter**. IF (MSRC001_0015[EffFreqReadOnlyLock]) THEN Read-only; Updated-by-hardware. ELSE Read-write; Updated-by-hardware. ENDIF. Incremented by hardware at the P0 frequency while the core is in C0. This register does not increment when the core is in the stop-grant state. In combination with MSRC000_00E8, this is used to determine the effective frequency of the core. A read of this MSR in guest mode is affected by MSRC000_0104 [Time Stamp Counter Ratio (TscRateMsr)]. This field uses software P-state numbering. See MSRC001_0015[EffFreqCntMwait], 2.5.2.3 [Effective Frequency], and 2.5.2.1.1.1 [Software P-state Numbering]. This register is not affected by writes to MSR0000_00E7.

MSRC000_00E8 Read-Only Actual Performance Frequency Clock Count (APerfReadOnly)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**APerfReadOnly: read-only actual core clocks counter**. IF (MSRC001_0015[EffFreqReadOnlyLock]) THEN Read-only; Updated-by-hardware. ELSE Read-write; Updated-by-hardware. ENDIF. This register increments in proportion to the actual number of core clocks cycles while the core is in C0. The register does not increment when the core is in the stop-grant state. See MSRC000_00E7. This register is not affected by writes to MSR0000_00E8.

MSRC000_0100 FS Base (FS_BASE)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**FSBase: expanded FS segment base**. Read-write; Not-same-for-all. This register provides access to the expanded 64-bit FS segment base. The address stored in this register must be in canonical form (if not canonical, a #GP fault fill occurs).

MSRC000_0101 GS Base (GS_BASE)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**GSBase: expanded GS segment base**. Read-write; Not-same-for-all. This register provides access to the expanded 64-bit GS segment base. The address stored in this register must be in canonical form (if not canonical, a #GP fault fill occurs).

MSRC000_0102 Kernel GS Base (KernelGSbase)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**KernelGSBase: kernel data structure pointer**. Read-write. This register holds the kernel data structure pointer which can be swapped with the GS_BASE register using the SwapGS instruction. The address stored in this register must be in canonical form (if not canonical, a #GP fault occurs).

MSRC000_0103 Auxiliary Time Stamp Counter (TSC_AUX)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:32	Reserved.
31:0	**TscAux: auxiliary time stamp counter data**. Read-write. It is expected that this is initialized by privileged software to a meaningful value, such as a processor ID. This value is returned in the RDTSCP instruction.

MSRC000_0104 Time Stamp Counter Ratio (TscRateMsr)

MSRC000_0104 [Time Stamp Counter Ratio (TscRateMsr)] allows the hypervisor to control the guest's view of the Time Stamp Counter. It provides a multiplier that scales the value returned when MSR0000_0010[TSC], MSR0000_00E7[MPERF], and MSRC000_00E7[MPerfReadOnly] are read by a guest running under virtualization. This allows the hypervisor to provide a consistent TSC, MPERF, and MPerfReadOnly rate for a guest process when moving that process between cores that have a differing P0 rate. The TSC Ratio MSR does not affect the value read from the TSC, MPERF, and MPerfReadOnly MSRs when read when in host mode or when virtualization is not being used or when accessed by code executed in system management mode (SMM) unless the SMM code is executed within a guest container. The TSC Ratio value does not affect the rate of the underlying TSC, MPERF, and MPerfReadOnly counters, or the value that gets written to the TSC, MPERF, and MPerfReadOnly MSRs counters on a write by either the host or the guest. The TSC Ratio MSR contains a

fixed-point number in 8.32 format, which is 8 bits of integer and 32 bits of fraction. This number is the ratio of the desired P0 frequency to the P0 frequency of the core. The reset value of the TSC Ratio MSR is 1.0, which results in a guest frequency matches the core P0 frequency.

Bits	Description
63:40	MBZ.
39:32	**TscRateMsrInt: time stamp counter rate integer**. Read-write. Reset: 01h. Specifies the integer part of the MSR TSC ratio value.
31:0	**TscRateMsrFrac: time stamp counter rate fraction**. Read-write. Reset: 0000_0000h. Specifies the fractional part of the MSR TSC ratio value.

MSRC000_0105 Lightweight Profile Configuration (LWP_CFG)

Bits	Description
63:48	Reserved.
47:40	**LwpVector: threshold interrupt vector**. Read-write. Reset: 0. Interrupt vector number used by LWP Threshold interrupts. Must be provided if LwpInt is set to 1.
39:32	**LwpCoreId: core ID**. Read-write; Not-same-for-all. Reset: 0. Core identification stored into the trace record. BIOS: CPUID Fn0000_0001_EBX[LocalApicId]. Software is recommended to set this to CPUID Fn0000_0001_EBX[LocalApicId].
31	**LwpInt: interrupt on threshold overflow**. Read-write. Reset: 0. 1=Enable LWP to interrupt on threshold overflow. CPUID Fn8000_001C_EAX[LwpInt] is an alias of MSRC000_0105[LwpInt].
30	**LwpPTSC: performance time stamp counter in event record**. Read-write. Reset: 0. 1=Enable storing performance time stamp in event record. CPUID Fn8000_001C_EAX[LwpPTSC] is an alias of MSRC000_0105[LwpPTSC].
29	**LwpCont: sampling in continuous mode**. Read-write. Reset: 0. 1=Enable continuous mode. 0=Enable synchronized mode. CPUID Fn8000_001C_EAX[LwpCont] is an alias of MSRC000_0105[LwpCont].
28:7	MBZ.
6	**LwpRNH: core reference clocks not halted event support**. MBZ. Reset: 0. 1=Enable LWP to count core reference clocks not halted. CPUID Fn8000_001C_EAX[LwpRNH] is an alias of MSRC000_0105[LwpRNH].
5	**LwpCNH: core clocks not halted event support**. MBZ. Reset: 0. 1=Enable LWP to count core clocks not halted. CPUID Fn8000_001C_EAX[LwpCNH] is an alias of MSRC000_0105[LwpCNH].
4	**LwpDME: DC miss event support**. MBZ. Reset: 0. 1=Enable LWP to count DC misses. CPUID Fn8000_001C_EAX[LwpDME] is an alias of MSRC000_0105[LwpDME].
3	**LwpBRE: branch retired event support**. Read-write. Reset: 0. 1=Enable LWP to count branches retired. CPUID Fn8000_001C_EAX[LwpBRE] is an alias of MSRC000_0105[LwpBRE].
2	**LwpIRE: instructions retired event support**. Read-write. Reset: 0. 1=Enable LWP to count instructions retired. CPUID Fn8000_001C_EAX[LwpIRE] is an alias of MSRC000_0105[LwpIRE].
1	**LwpVAL: LWPVAL instruction support**. Read-write. Reset: 0. 1=LWPVAL instruction is enabled. CPUID Fn8000_001C_EAX[LwpVAL] is an alias of MSRC000_0105[LwpVAL].
0	Reserved.

MSRC000_0106 Lightweight Profile Control Block Address (LWP_CBADDR)

Access to the internal copy of the LWPCB logical line/64 B address. A read returns the current LWPCB address without performing any of the operations described for the SLWPCB instruction. A write to this register with a non-zero value causes a #GP fault. Use LLWPCB or XRSTOR to load an LWPCB address. Writing a zero to LWP_CBADDR will immediately disable LWP, discarding any internal state. For instance, an operating system can write a zero to stop LWP when it terminates a thread. All references to the LWPCB implicitly use the DS segment register. Must be 64 B aligned.

Bits	Description
63:6	**LwpCbAddr[63:6]: control block logical address**. Read-write. Reset: 0. LwpCbAddr[63:0] = {LwpCbAddr[63:6], 000000b}.
5:0	RAZ.

MSRC000_0408 NB Machine Check Misc 4 (Link Thresholding) 1 (MC4_MISC1)

Per-node. MSRC000_0408 is associated with the link error type. See 2.16.1.7 [Error Thresholding]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn].

Bits	Description
63	**Valid**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn]) THEN Read-write; Per-node; Not-same-for-all. ELSE Read-only; Per-node; Not-same-for-all. Reset: 1. ENDIF. 1=The CntP field is present.
62	**CntP: counter present**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn]) THEN Read-write; Per-node; Not-same-for-all. ELSE Read-only; Per-node; Not-same-for-all. Reset: 1. ENDIF. 1=A valid threshold counter is present.
61	**Locked**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn]) THEN Read-write; Per-node; not-same-for-all. ELSE Read-only; Per-node; not-same-for-all. Reset: 0. ENDIF. BIOS: IF (IntType == 10b) THEN 1. ELSE 0. ENDIF. 1=Writes to bits[55:32] of this register are ignored. Set by BIOS to indicate that this register is not available for OS use. When MSRC001_0015[McStatusWrEn] is set, MSR writes to this register update all bits, regardless of the state of the Locked bit.
60	**IntP**. Read-only. Reset: 1.

59:56	Reserved.
55:52	**LvtOffset: LVT offset**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn] \| ~MSRC000_0408[Locked]) THEN Read-write; Per-node; Not-same-for-all. ELSE Read-only; Per-node; Not-same-for-all. Reset: 0. ENDIF. Specifies the address of the LVT entry in the APIC registers as follows: LVT address = (LvtOffset shifted left 4 bits) + 500h (see APIC[530:500]). Bits Description 3h-0h See APIC[530:500]. Fh-4h Reserved.
51	**CntEn: counter enable**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn] \| ~Locked) THEN Read-write; Per-node; Not-same-for-all. ELSE Read-only; Per-node; Not-same-for-all. Reset: 0. ENDIF. 1=Count thresholding errors. See 2.16.1.7 [Error Thresholding].
50:49	**IntType: interrupt type**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn] \| ~MSRC000_0408[Locked]) THEN Read-write; Per-node; Not-same-for-all. ELSE Read-only; Per-node; Not-same-for-all. Cold reset: 0. ENDIF. Specifies the type of interrupt signaled when Ovrflw is set. Bits Description 00b No Interrupt. 01b APIC. APIC based interrupt (see LvtOffset above) to all cores. 10b SMI. SMI trigger event (always routed to CpuCoreNum 0, as defined in 2.4.4 [Processor Cores and Downcoring]); see 2.4.9.2.3 [SMI Sources And Delivery]. 11b Reserved.
48	**Ovrflw: overflow.** IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn] \| ~Locked) THEN Read-write; Per-node; Not-same-for-all; Set-by-hardware. ELSE Read-only; Per-node; Not-same-for-all; Set-by-hardware. Cold reset: 0. ENDIF. Set by hardware when ErrCnt transitions from FFEh to FFFh. When this bit is set, the interrupt selected by the IntType field is generated.
47:44	Reserved.
43:32	**ErrCnt: error counter**. IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ; Per-node; Not-same-for-all. ELSEIF (MSRC001_0015[McStatusWrEn] \| ~Locked) THEN Read-write; Per-node; Not-same-for-all; Updated-by-hardware. ELSE Read-only; Per-node; Not-same-for-all; Updated-by-hardware. Cold reset: 0. ENDIF. Written by software to set the starting value of the error counter. Incremented by hardware when errors are logged. Saturates at FFFh (no rollover). The threshold value, written by software, is (FFFh - the desired error count (the number of errors necessary in order for an interrupt to be taken)).
31:0	Reserved.

MSRC000_0409 NB Machine Check Misc 4 (L3 Thresholding) 1 (MC4_MISC2)

Bits	Description
63:32	Alias of D18F3x170.
31:0	Reserved.

MSRC000_040[F:A] Reserved

Bits	Description
63:0	RAZ.

MSRC000_0410 Machine Check Deferred Error Configuration (CU_DEFER_ERR)

Read-write; Per-compute-unit; Same-for-all. Configures deferred errors for all MC banks.

Bits	Description
63:8	Reserved.
7:4	**DeferredLvtOffset: NB and core deferred LVT offset**. Read-write. Reset: 0. BIOS: 2h. See MSR0000_0413[LvtOffset].
3	Reserved.
2:1	**DeferredIntType: NB and core deferred interrupt type**. Read-write. Reset: 0. See MSR0000_0413[IntType]. Bits — Description 00b — No Interrupt. 01b — APIC. APIC based interrupt (see LvtOffset above) to all cores. 10b — SMI. SMI trigger event (always routed to CpuCoreNum 0, as defined in 2.4.4 [Processor Cores and Downcoring]); see 2.4.9.2.3 [SMI Sources And Delivery]. 11b — Reserved.
0	Reserved.

3.22 MSRs - MSRC001_0xxx

MSRC001_00[03:00] Performance Event Select (PERF_CTL[3:0])

Reset: 0000_0000_0000_0000h. See 2.6.1 [Performance Monitor Counters].
The legacy alias of MSRC001_020[6,4,2,0]. See MSRC001_020[A,8,6,4,2,0].

Table 257: Register Mapping for MSRC001_00[03:00]

Register	Function
MSRC001_0000	Counter 0
MSRC001_0001	Counter 1
MSRC001_0002	Counter 2
MSRC001_0003	Counter 3

Bits	Description
63:0	MSRC001_00[03:00] is an alias of MSRC001_020[6,4,2,0].

MSRC001_00[07:04] Performance Event Counter (PERF_CTR[3:0])

The legacy alias of MSRC001_020[7,5,3,1]. See MSRC001_020[B,9,7,5,3,1].

Table 258: Register Mapping for MSRC001_00[07:04]

Register	Function
MSRC001_0004	Counter 0
MSRC001_0005	Counter 1
MSRC001_0006	Counter 2
MSRC001_0007	Counter 3

Bits	Description
63:0	MSRC001_00[07:04] is an alias of MSRC001_020[7,5,3,1].

MSRC001_0010 System Configuration (SYS_CFG)

Bits	Description
63:23	Reserved.
22	**Tom2ForceMemTypeWB: top of memory 2 memory type write back**. Read-write; Per-compute-unit. Reset: 0. 1=The default memory type of memory between 4GB and TOM2 is write back instead of the memory type defined by MSR0000_02FF[MemType]. For this bit to have any effect, MSR0000_02FF[MtrrDefTypeEn] must be 1. MTRRs and PAT can be used to override this memory type.
21	**MtrrTom2En: MTRR top of memory 2 enable**. Read-write; Per-compute-unit. Reset: 0. 0=MSRC001_001D [Top Of Memory 2 (TOM2)] is disabled. 1=This register is enabled. See D0F0x64_x19[TomEn].

20	**MtrrVarDramEn: MTRR variable DRAM enable**. Read-write; Per-compute-unit. Reset: 0. BIOS: 1. 0=MSRC001_001A [Top Of Memory (TOP_MEM)] and IORRs are disabled. 1=These registers are enabled.
19	**MtrrFixDramModEn: MTRR fixed RdDram and WrDram modification enable**. Read-write. Reset: 0. Controls access to MSR0000_02[6F:68,59:58,50][RdDram, WrDram]. 0=Access type is MBZ; writing 00b does not change the hidden value of MSR0000_02[6F:68,59:58,50][RdDram, WrDram]. 1=Access type is Read-write. BIOS: This bit should be set to 1 during BIOS initialization of the fixed MTRRs, then cleared to 0 for operation.
18	**MtrrFixDramEn: MTRR fixed RdDram and WrDram attributes enable**. Read-write; Per-compute-unit. Reset: 0. BIOS: 1. 1=Enables the RdDram and WrDram attributes in MSR0000_02[6F:68,59:58,50].
17	Reserved.
16	**ChgToDirtyDis: change to dirty disable**. Read-write; Per-compute-unit. Reset: 0. 1=Disables Change-to-Dirty command; The change-to-dirty condition is handled by evicting the line and then fetching it with a RdBlkM command.
15:0	Reserved.

MSRC001_0015 Hardware Configuration (HWCR)

Bits	Description
63:30	Reserved.
29	**CSEnable: connected/modern standby enable**. Read-write. Reset: 0. 0=Connected/Modern standby feature is disabled. No Local APIC writes to NB PCI space or C6 save space occurs on C6 entry. No Local APIC restore from C6 save space occurs on C6 exit. No C6 state save skip will occur. 1=Connected/Modern standby feature is enabled. Local APIC writes to NB PCI Space and C6 save space can occur if MSR0000_001B[ApicEn] is set. Local APIC restore from C6 save space on C6 exit can occur if MSR0000_001B[ApicEn].
28	**SmuLock**. Read-write. Reset: 0. BIOS: 1.
27	**EffFreqReadOnlyLock: read-only effective frequency counter lock**. Write-1-only. Reset: 0. BIOS: 1. 1=MSRC000_00E7 and MSRC000_00E8 are read-only.
26	**EffFreqCntMwait: effective frequency counting during mwait**. Read-write. Reset: 0. Specifies whether MSR0000_00E7 [Max Performance Frequency Clock Count (MPERF)] and MSR0000_00E8 [Actual Performance Frequency Clock Count (APERF)] increment while the core is in the monitor event pending state. 0=The registers do not increment. 1=The registers increment. See 2.5.2.3 [Effective Frequency].
25	**CpbDis: core performance boost disable**. Read-write. Reset: 0. Specifies whether core performance boost is requested to be enabled or disabled. 0=CPB is requested to be enabled. 1=CPB is disabled. See 2.5.8 [Application Power Management (APM)]. If core performance boost is disabled while a core is in a boosted P-state, the core automatically transitions to the highest performance non-boosted P-state.
24	**TscFreqSel: TSC frequency select**. Read-only. Reset: 1. 1=The TSC increments at the P0 frequency. This field uses software P-state numbering. See 2.5.2.1.1.1 [Software P-state Numbering].
23	**ForceRdWrSzPrb: force probes for RdSized and WrSized**. Read-write. Reset: 0. A read returns a 1 if this field is set on any core of the node. 1=Forces probes on read-sized and write-sized transactions, except those that are display refresh.

22:21	Reserved.
20	**IoCfgGpFault: IO-space configuration causes a GP fault**. Read-write. Reset: 0. 1=IO-space accesses to configuration space cause a GP fault. The fault is triggered if any part of the IO read/write address range is between CF8h and CFFh, inclusive. These faults only result from single IO instructions, not to string and REP IO instructions. This fault takes priority over the IO trap mechanism described by MSRC001_0054 [IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)].
19	Reserved.
18	**McStatusWrEn: machine check status write enable**. Read-write. Reset: 0. McStatusWrEn can be used to debug machine check exception and interrupt handlers. See 2.16.1 [Machine Check Architecture]. • 1=MCi_STATUS registers are read-write, including reserved fields; do not cause general protection faults; such writes update all implemented bits in these registers; All fields of all threshold registers are Read-write when accessed from MSR space, including Locked, except BlkPtr which is always read-only; McStatusWrEn does not change the access type for the thresholding registers accessed via configuration space. • 0=MCi_STATUS registers are readable; writing a non-zero pattern to these registers causes a general protection fault. • The MCi_STATUS registers are: MSR0000_0401, MSR0000_0405, MSR0000_0409, MSR0000_040D, MSR0000_0411, MSR0000_0415, MSR0000_0419. McStatusWrEn does not affect the writability of MSR0000_0001; MSR0000_0001 is always writable. • The thresholding registers affected by McStatusWrEn are: MSR0000_0403, MSR0000_0407, MSR0000_040B, MSR0000_0413, MSR0000_0417, MSRC000_0408.
17	**Wrap32Dis: 32-bit address wrap disable**. Read-write. Reset: 0. 1=Disable 32-bit address wrapping. Software can use Wrap32Dis to access physical memory above 4 Gbytes without switching into 64-bit mode. To do so, software should write a greater-than 4 Gbyte address to MSRC000_0100 [FS Base (FS_BASE)] and MSRC000_0101 [GS Base (GS_BASE)]. Then it would address ±2 Gbytes from one of those bases using normal memory reference instructions with a FS or GS override prefix. However, the INVLPG, FST, and SSE store instructions generate 32-bit addresses in legacy mode, regardless of the state of Wrap32Dis.
16:15	Reserved.
14	**RsmSpCycDis: RSM special bus cycle disable**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 0=A link special bus cycle, SMIACK, is generated on a resume from SMI.
13	**SmiSpCycDis: SMI special bus cycle disable**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. 0=A link special bus cycle, SMIACK, is generated when an SMI interrupt is taken.
12	**HltXSpCycEn: halt-exit special bus cycle enable**. Read-write. Reset: 0. BIOS: 1. Read-write. Specifies whether a halt exit special bus cycle is sent to the Northbridge when exiting from the halt state. 1=Messages are sent. 0=Messages are not sent. See 2.5.2.2.4.1 [FCH Messaging].
11	Reserved.
10	**MonMwaitUserEn: MONITOR/MWAIT user mode enable**. Read-write. Reset: 0. 1=The MONITOR and MWAIT instructions are supported in all privilege levels. 0=The MONITOR and MWAIT instructions are supported only in privilege level 0; these instructions in privilege levels 1 to 3 cause a #UD exception. The state of this bit is ignored if MonMwaitDis is set.
9	**MonMwaitDis: MONITOR and MWAIT disable**. Read-write. Reset: 0. 1=The MONITOR and MWAIT opcodes become invalid. This affects what is reported back through CPUID Fn0000_0001_ECX[Monitor].

8	**IgnneEm: IGNNE port emulation enable**. Read-write. Reset: 0. 1=Enable emulation of IGNNE port.
7	**AllowFerrOnNe: allow FERR on NE**. Read-write. Reset: 0. IF ((AllowFerrOnNe == 1) && (CR0.NE == 1)) THEN disable legacy FERR signaling and generate FERR exception directly. ENDIF. 0=Legacy FERR signlling.
6:5	Reserved.
4	**INVDWBINVD: INVD to WBINVD conversion**. Read-write. Reset: 1. 1=Convert INVD to WBINVD. BIOS: See 2.3.3 [Using L2 Cache as General Storage During Boot]. This bit is required to be set for normal operation when both cores of a compute unit are enabled, and thus share the L2 cache.
3	**TlbCacheDis: cacheable memory disable**. Read-write. Reset: 0. 1=Disable performance improvement that assumes that the PML4, PDP, PDE and PTE entries are in cacheable WB DRAM. Operating systems that maintain page tables in any other memory type must set the TlbCacheDis bit to insure proper operation. • TlbCacheDis does not override the memory type specified by the SMM ASeg and TSeg memory regions. See 2.4.9.2.7 [The Protected ASeg and TSeg Areas].
2:1	Reserved.
0	**SmmLock: SMM code lock**. Read; Write-1-only. Reset: 0. SBIOS: 1. 1=SMM code in the ASeg and TSeg range and the SMM registers are read-only and SMI interrupts are not intercepted in SVM. See 2.4.9.2.9 [Locking SMM].

MSRC001_00[18,16] IO Range Base (IORR_BASE[1:0])

Per-compute-unit. Reset: X. MSRC001_0016 and MSRC001_0017 combine to specify the first IORR range and MSRC001_0018 and MSRC001_0019 combine to specify the second IORR range. A core access, with address CPUAddr, is determined to be within IORR address range if the following equation is true: CPUAddr[47:12] & PhyMask[47:12] == PhyBase[47:12] & PhyMask[47:12].
BIOS can use the IORRs to create an IO hole within a range of addresses that would normally be mapped to DRAM. It can also use the IORRs to re-assert a DRAM destination for a range of addresses that fall within a bigger IO hole that overlays DRAM. See 2.4.6.1.2 [Determining The Access Destination for Core Accesses].

Bits	Description
63:48	RAZ.
47:12	**PhyBase: physical base address**. Read-write.
11:5	RAZ.
4	**RdMem: read from memory**. Read-write. 1=Read accesses to the range are directed to system memory. 0=Read accesses to the range are directed to IO.
3	**WrMem: write to memory**. Read-write. 1=Write accesses to the range are directed to system memory. 0=Write accesses to the range are directed to IO.
2:0	RAZ.

MSRC001_00[19,17] IO Range Mask (IORR_MASK[1:0])

Per-compute-unit.
Reset: 0000_0000_0000_0000h. See MSRC001_00[18,16].

Bits	Description
63:48	RAZ.
47:12	**PhyMask: physical address mask**. Read-write.
11	**Valid**. Read-write. 1=The pair of registers that specifies an IORR range is valid.
10:0	RAZ.

MSRC001_001A Top Of Memory (TOP_MEM)

Per-compute-unit.
Reset: 0000_0000_0000_0000h.

Bits	Description
63:48	RAZ.
47:23	**TOM[47:23]: top of memory**. Read-write. Specifies the address that divides between MMIO and DRAM. This value is normally placed below 4G. From TOM to 4G is MMIO; below TOM is DRAM. See 2.4.6 [System Address Map] and 2.9.13 [DRAM CC6/PC6 Storage].
22:0	RAZ.

MSRC001_001D Top Of Memory 2 (TOM2)

Per-compute-unit.
Reset: 0000_0000_0000_0000h.

Bits	Description
63:48	RAZ.
47:23	**TOM2[47:23]: second top of memory**. Read-write. Specifies the address divides between MMIO and DRAM. This value is normally placed above 4G. From 4G to (TOM2 - 1) is DRAM; TOM2 and above is MMIO. See 2.4.6 [System Address Map] and 2.9.13 [DRAM CC6/PC6 Storage]. This register is enabled by MSRC001_0010[MtrrTom2En].
22:0	RAZ.

MSRC001_001F Northbridge Configuration 1 (NB_CFG1)

Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. MSRC001_001F[31:0] is an alias of D18F3x88. MSRC001_001F[63:32] is an alias of D18F3x8C.

Bits	Description
63:32	Alias of D18F3x8C.
31:0	Alias of D18F3x88.

MSRC001_0022 Machine Check Exception Redirection

Reset: 0000_0000_0000_0000h. This register can be used to redirect machine check exceptions (MCEs) to SMIs or vectored interrupts. If both RedirSmiEn and RedirVecEn are set, then undefined behavior results.

Bits	Description
63:10	Reserved.
9	**RedirSmiEn**. Read-write. 1=Redirect MCEs (that are directed to this core) to generate an SMI-trigger IO cycle via MSRC001_0056. The status is stored in SMMFEC4[MceRedirSts].
8	**RedirVecEn**. Read-write. 1=Redirect MCEs (that are directed to this core) to generate a vectored interrupt, using the interrupt vector specified in RedirVector.
7:0	**RedirVector**. Read-write. See RedirVecEn.

MSRC001_00[35:30] Processor Name String

Reset: 0000_0000_0000_0000h. BIOS: Table 260. These registers holds the CPUID name string in ASCII. The state of these registers are returned by CPUID instructions, CPUID Fn8000_000[4:2]_E[D,C,B,A]X. BIOS should set these registers to the product name for the processor as provided by AMD. Each register contains a block of 8 ASCII characters; the least byte corresponds to the first ASCII character of the block; the most-significant byte corresponds to the last character of the block. MSRC001_0030 contains the first block of the name string; MSRC001_0035 contains the last block of the name string.

Table 259: Register Mapping for MSRC001_00[35:30]

Register	Function
MSRC001_0030	Characters 7-0
MSRC001_0031	Characters 15-8
MSRC001_0032	Characters 23-16
MSRC001_0033	Characters 31-24
MSRC001_0034	Characters 39-32
MSRC001_0035	Characters 47-40

See D0F0xBC_xD021_1088 for the access method to D0F0xBC_xD021_108C.

Table 260: BIOS Recommendations for MSRC001_00[35:30]

Register	BIOS
MSRC001_0030	{D0F0xBC_xD021_108C_x1, D0F0xBC_xD021_108C_x0}
MSRC001_0031	{D0F0xBC_xD021_108C_x3, D0F0xBC_xD021_108C_x2}
MSRC001_0032	{D0F0xBC_xD021_108C_x5, D0F0xBC_xD021_108C_x4}
MSRC001_0033	{D0F0xBC_xD021_108C_x6, D0F0xBC_xD021_108C_x6}
MSRC001_0034	{D0F0xBC_xD021_108C_x9, D0F0xBC_xD021_108C_x8}
MSRC001_0035	{D0F0xBC_xD021_108C_xB, D0F0xBC_xD021_108C_xA}

Bits	Description
63:0	**CpuNameString**. Read-write.

MSRC001_003E Hardware Thermal Control (HTC)

Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. Uses hardware P-state numbering. See 2.5.2.1.1.2 [Hardware P-state Numbering].

Bits	Description
63:32	Reserved.
31:0	Alias of D0F0xBC_xD820_0C64.

MSRC001_0044 DC Machine Check Control Mask (MC0_CTL_MASK)

Read-write. Reset: 0000_0000_0000_0000h. BIOS: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSR0000_0400 [MC0 Machine Check Control (MC0_CTL)].

Bits	Description
63:12	Reserved.
11	**MAB: Miss Address Buffer parity**.
10	**HWA: hardware assert**.
9	Reserved.
8	**IntErrType2: internal error type 2**.
7	**SRDE: read data errors**. System read data errors on cache fill.
6	**LFE: line fill error**. Uncorrectable error on cache fill.
5	**SCBP: SCB parity**.
4	**SQP: store queue parity**.
3	**LQP: load queue parity**.
2	**DatP: data parity**.
1	**TLBP: TLB parity**.
0	**TagP: tag parity**.

MSRC001_0045 IC Machine Check Control Mask (MC1_CTL_MASK)

Read-write; Per-compute-unit. Reset: 0000_0000_0000_0080h. BIOS: 0000_0000_0000_0080h. See 2.16.1 [Machine Check Architecture]. See MSR0000_0404 [MC1 Machine Check Control (MC1_CTL)].

Bits	Description
63:26	Reserved.
25	**HWA: Hardware assertion**.
24	**DEDISPQ: decoder dispatch micro-op queue parity error**.
23	**IVP: IC valid bit parity error**.
22	**L1TLBM: IC L1 TLB multi-match error**.
21	**L2TLBM: IC L2 TLB multi-match error**.
20	**DFIFOE: decoder FIFO parity error**.
19	**DPDBE: decoder predecode buffer parity error**.

18	**DEIBP: decoder instruction buffer parity error**.
17	**DEUOPQP: Decoder micro-op queue parity error**.
16	**DEPRP: microcode patch buffer parity error**.
15	**BSRP: branch status register parity error**.
14	Reserved.
13	**PQP: prediction queue parity error**.
12	**PFBP: prefetch buffer parity**.
11:10	Reserved.
9	**SRDE: system read data error**.
8	Reserved.
7	**LineFillPoison: line fill poison error**.
6	**L1TP: L1 TLB parity error**.
5	**L2TP: L2 TLB parity error**.
4	**ISTP: L1 cache probe tag array parity error**.
3	**IMTP: L1 cache main tag array parity error**.
2	**IDP: L1 cache data array parity error**.
1:0	Reserved.

MSRC001_0046 BU Machine Check Control Mask (MC2_CTL_MASK)

Read-write; Per-compute-unit. Reset: 0000_0000_0000_0000h. BIOS: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSR0000_0408 [MC2 Machine Check Control (MC2_CTL)].

Bits	Description
63:16	Reserved.
15	**HWA: Hardware assertion**.
14	**L2TlbFill: TLB fill error enable**. Data with uncorrectable error provided to TLB.
13	**RdData: read data error from NB**.
12	**L2Tag: L2 cache tag error**.
11	**L2TlbData: L2 TLB parity error**. Parity error reading from TLB.
10	**L2Prefetch: L2 data prefetcher parity error**.
9	**XabAddr: XAB address parity error**.
8	**PrbAddr: probe buffer address parity error**.
7	**FillData: fill data parity and ECC error**.
6	**PrqAddr: post retire queue address parity error**.
5	**PrqData: post retire queue data parity error**.
4	**WccAddr: write coalescing cache address ECC error**.
3	**WccData: write coalescing cache data ECC error**.
2	**WcbData: write combining buffer data parity error**.

1	**VbData: victim buffer data parity and ECC error**.
0	**L2TagMultiHit: L2 tag multiple hit error**.

MSRC001_0047 Reserved (MC3_CTL_MASK)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	Reserved.

MSRC001_0048 NB Machine Check Control Mask (MC4_CTL_MASK)

IF (D18F3x44[NbMcaToMstCpuEn] && !NBC) THEN RAZ. ELSE Read-write; Per-node. ENDIF. The format of MC4_CTL_MASK corresponds to MSR0000_0410 [MC4 Machine Check Control (MC4_CTL)]. For each defined bit position, 1=Disable logging. MCi_CTL_MASK allow BIOS to mask the presence of any error source from software for test and debug. When Error Sources are masked, it is as if the error was not detected. Such masking consequently prevents error responses and actions. Only one of these registers exists in multi-core devices; see 3.1.1 [Northbridge MSRs In Multi-Core Products]. See 2.16.1 [Machine Check Architecture]. Accessibility of this register by non-NBC cores is affected by D18F3x44[NbMcaToMstCpuEn]. See MSR0000_0410 [MC4 Machine Check Control (MC4_CTL)].

Bits	Description
63:32	Reserved.
31:0	Alias of D18F3x198.

MSRC001_0049 EX Machine Check Control Mask (MC5_CTL_MASK)

Reset: 0000_0000_0000_0000h. BIOS: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSR0000_0414 [MC5 Machine Check Control (MC5_CTL)].

Bits	Description
63:0	See: MSR0000_0414. The format of MC5_CTL_MASK corresponds to MC5_CTL.

MSRC001_004A FP Machine Check Control Mask (MC6_CTL_MASK)

Per-compute-unit. Reset: 0000_0000_0000_0000h. BIOS: 0000_0000_0000_0000h. See 2.16.1 [Machine Check Architecture]. See MSR0000_0418 [MC6 Machine Check Control (MC6_CTL)].

Bits	Description
63:0	See: MSR0000_0418. The format of MC6_CTL_MASK corresponds to MC6_CTL.

MSRC001_00[53:50] IO Trap (SMI_ON_IO_TRAP_[3:0])

Reset: 0000_0000_0000_0000h.

MSRC001_00[53:50] and MSRC001_0054 provide a mechanism for executing the SMI handler if a an access to one of the specified addresses is detected. Access address and access type checking is performed before IO instruction execution. If the access address and access type match one of the specified IO address and access types, then: (1) the IO instruction is not executed; (2) any breakpoint, other than the single-step breakpoint, set

on the IO instruction is not taken (the single-step breakpoint is taken after resuming from SMM); and (3) the SMI-trigger IO cycle specified by MSRC001_0056. The status is stored in SMMFEC4[IoTrapSts].

IO-space configuration accesses are special IO accesses. An IO access is defined as an IO-space configuration access when IO instruction address bits[31:0] are CFCh, CFDh, CFEh, or CFFh when IO-space configuration is enabled (IOCF8[ConfigEn]). The access address for a configuration space access is the current value of IOCF8[BusNo, Device, Function, RegNo]. The access address for an IO access that is not a configuration access is equivalent to the IO instruction address, bits[31:0].

The access address is compared with SmiAddr, and the instruction access type is compared with the enabled access types defined by ConfigSMI, SmiOnRdEn, and SmiOnWrEn. Access address bits[23:0] can be masked with SmiMask.

IO and configuration space trapping to SMI applies only to single IO instructions; it does not apply to string and REP IO instructions.

The conditional GP fault described by MSRC001_0015[IoCfgGpFault] takes priority over this trap.

Table 261: Register Mapping for MSRC001_00[53:50]

Register	Function
MSRC001_0050	Range 0
MSRC001_0051	Range 1
MSRC001_0052	Range 2
MSRC001_0053	Range 3

Bits	Description
63	**SmiOnRdEn: enable SMI on IO read**. Read-write. 1=Enables SMI generation on a read access.
62	**SmiOnWrEn: enable SMI on IO write**. Read-write. 1=Enables SMI generation on a write access.
61	**ConfigSmi: configuration space SMI**. Read-write. 1=Configuration access. 0=IO access (that is not an IO-space configuration access).
60:56	Reserved.
55:32	**SmiMask[23:0]**. Read-write. SMI IO trap mask. 0=Mask address bit. 1=Do not mask address bit.
31:0	**SmiAddr[31:0]**. Read-write. SMI IO trap address.

MSRC001_0054 IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)

For each of the SmiEn bits below, 1=The trap specified by the corresponding MSR is enabled. See MSRC001_00[53:50].

Bits	Description
63:32	RAZ.
31:16	Reserved.
15	**IoTrapEn: IO trap enable**. Read-write. Reset: 0. 1=Enable IO and configuration space trapping specified by MSRC001_00[53:50] and MSRC001_0054.
14:8	Reserved.

7	**SmiEn3: SMI enable for the trap specified by MSRC001_0053**. Read-write. Reset: 0.
6	Reserved.
5	**SmiEn2: SMI enable for the trap specified by MSRC001_0052**. Read-write. Reset: 0.
4	Reserved.
3	**SmiEn1: SMI enable for the trap specified by MSRC001_0051**. Read-write. Reset: 0.
2	Reserved.
1	**SmiEn0: SMI enable for the trap specified by MSRC001_0050**. Read-write. Reset: 0.
0	Reserved.

MSRC001_0055 Interrupt Pending

This register is used to specify messages that the processor generates under certain conditions, that target the IO hub. One purpose is to ensure that the IO hub can wake the processor out of the stop-grant state when there is a pending interrupt. Otherwise, it is possible for the processor to remain in the stop-grant state while an interrupt is pending in the processor. This is accomplished by sending a message to the IO hub to indicate that the interrupt is pending. There are two message types: a programmable IO-space message and the link INT_PENDING message defined by the link specification.

If the IO hub does not support the INT_PENDING message, the IO space message should be selected by IntPndMsg. When this is enabled, the check for a pending interrupt is performed at the end of each IO instruction. If there is a pending interrupt and STPCLK is asserted, the processor executes a byte-size IO access as specified by IORd, IOMsgAddr, and IOMsgData.

If the IO hub supports the INT_PENDING message, it should be selected by IntPndMsg. The check for a pending interrupt is performed while in the stop-grant state or when entering the stop-grant state. If there is a pending interrupt, the processor broadcasts the INT_PENDING message. An INT_PENDING message may not be generated for arbitrated interrupts in multi-node systems.

Bits	Description
63:32	RAZ.
31	Reserved.
30	**EnablePmTmrCheckLoop**. Read-write. Reset: 0. 1=The core loops on IO-space read accesses to the address specified by IOMsgAddr until the data value has incremented from the previous read access.
29:27	Reserved.
26	**IORd: IO Read**. Read-write. Reset: 0. 1=IO read. 0=IO write.
25	**IntPndMsg: interrupt pending message**. Read-write. Reset: 0. Selects the interrupt pending message type. 0=Link-defined INT_PENDING message; 1=Programmable SMI-trigger IO-space message. The status is stored in SMMFEC4[IntPendSmiSts].
24	**IntPndMsgDis: interrupt pending message disable**. Read-write. Reset: 0. Disable generating the interrupt pending message specified by IntPndMsg.
23:16	**IOMsgData: IO message data**. Read-write. Reset: 0. IO write message data. This field is only used if IORd specifies an IO write message.
15:0	**IOMsgAddr: IO message address**. Read-write. Reset: 0. IO space message address.

MSRC001_0056 SMI Trigger IO Cycle

Not-same-for-all. Reset: 0000_0000_0000_0000h. See 2.4.9.2.3 [SMI Sources And Delivery]. This register specifies an IO cycle that may be generated when a local SMI trigger event occurs. If IoCycleEn is set and there is a local SMI trigger event, then the IO cycle generated is a byte read or write, based on IoRd, to address IoPortAddress. If the cycle is a write, then IoData contains the data written. If the cycle is a read, the value read is discarded. If IoCycleEn is clear and a local SMI trigger event occurs, then undefined behavior results.

Bits	Description
63:32	Reserved.
31:27	Reserved.
26	**IoRd: IO Read**. Read-write. 1=IO read; 0=IO write.
25	**IoCycleEn: IO cycle enable**. Read-write. 1=The SMI trigger IO cycle is enabled to be generated.
24	Reserved.
23:16	**IoData**. Read-write.
15:0	**IoPortAddress**. Read-write.

MSRC001_0058 MMIO Configuration Base Address

Same-for-all. See 2.7 [Configuration Space] for a description of MMIO configuration space.

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>63:48</td><td>RAZ.</td></tr>
<tr><td>47:20</td><td>MmioCfgBaseAddr[47:20]: MMIO configuration base address bits[47:20]. Read-write. Reset: X. Specifies the base address of the MMIO configuration range.</td></tr>
<tr><td>19:6</td><td>RAZ.</td></tr>
<tr><td>5:2</td><td>BusRange: bus range identifier. Read-write. Reset: 0. Specifies the number of buses in the MMIO configuration space range. The size of the MMIO configuration space is 1 MB times the number of buses.
<table>
<tr><th>Bits</th><th>Description</th><th>Bits</th><th>Description</th></tr>
<tr><td>0h</td><td>1</td><td>5h</td><td>32</td></tr>
<tr><td>1h</td><td>2</td><td>6h</td><td>64</td></tr>
<tr><td>2h</td><td>4</td><td>7h</td><td>128</td></tr>
<tr><td>3h</td><td>8</td><td>8h</td><td>256</td></tr>
<tr><td>4h</td><td>16</td><td>Fh-9h</td><td>Reserved.</td></tr>
</table></td></tr>
<tr><td>1</td><td>Reserved.</td></tr>
<tr><td>0</td><td>Enable. Read-write. Reset: 0. 1=MMIO configuration space is enabled.</td></tr>
</table>

MSRC001_0060 BIST Results

Read; GP-write. Reset: 0000_0000_XXXX_XXXXh. This register provides BIST results after each reset. The results provided here are identical to the values provided in EAX.
If (MSRC001_0060[30:0] == 0000_0000h) then no BIST failures were detected.

Bits	Description
63:32	Reserved.
31:0	**BistResults**.

MSRC001_0061 P-state Current Limit

Read; GP-write; Per-compute-unit; Updated-by-hardware. See 2.5.2 [CPU Core Power Management].

Bits	Description
63:7	RAZ.
6:4	**PstateMaxVal: P-state maximum value**. Specifies the lowest-performance non-boosted P-state (highest non-boosted value) allowed. Attempts to change MSRC001_0062[PstateCmd] to a lower-performance P-state (higher value) are clipped to the value of this field. This field uses software P-state numbering. See 2.5.2.1.1.1 [Software P-state Numbering].
3	RAZ.
2:0	**CurPstateLimit: current P-state limit**. Specifies the highest-performance non-boosted P-state (lowest value) allowed. CurPstateLimit is always bounded by MSRC001_0061[PstateMaxVal]. Attempts to change the CurPstateLimit to a value greater (lower performance) than MSRC001_0061[PstateMaxVal] leaves CurPstateLimit unchanged. This field uses software P-state numbering. See MSRC001_0071[CurPstateLimit] and 2.5.2.1.1.1 [Software P-state Numbering].

MSRC001_0062 P-state Control

Bits	Description
63:3	MBZ.
2:0	**PstateCmd: P-state change command**. Read-write; Not-same-for-all. Cold reset value varies by product; after a warm reset, value initializes to the P-state the core was in prior to the reset. Writes to this field cause the core to change to the indicated non-boosted P-state number, specified by MSRC001_00[6B:64]. 0=P0, 1=P1, etc. P-state limits are applied to any P-state requests made through this register. Reads from this field return the last written value, regardless of whether any limits are applied. This field uses software P-state numbering. See 2.5.2 [CPU Core Power Management] and 2.5.2.1.1.1 [Software P-state Numbering].

MSRC001_0063 P-state Status

Read; GP-write; Per-compute-unit; Updated-by-hardware.

Bits	Description
63:3	RAZ.
2:0	**CurPstate: current P-state**. Cold reset: Varies by product. This field provides the frequency component of the current non-boosted P-state of the core (regardless of the source of the P-state change, including MSRC001_0062[PstateCmd]. 0=P0, 1=P1, etc. The value of this field is updated when the COF transitions to a new value associated with a P-state. This field uses software P-state numbering. See 2.5.2 [CPU Core Power Management] and 2.5.2.1.1.1 [Software P-state Numbering].

MSRC001_00[6B:64] P-state [7:0]

Per-node. Cold reset: Varies by product. Each of these registers specify the frequency and voltage associated with each of the core P-states.

Table 262: Register Mapping for MSRC001_00[6B:64]

Register	Function
MSRC001_0064	P-state 0
MSRC001_0065	P-state 1
MSRC001_0066	P-state 2
MSRC001_0067	P-state 3
MSRC001_0068	P-state 4
MSRC001_0069	P-state 5
MSRC001_006A	P-state 6
MSRC001_006B	P-state 7

The CpuVid field in these registers is required to be programmed to the same value in all cores of a processor, but are allowed to be different between processors in a multi-processor system. All other fields in these registers are required to be programmed to the same value in each core of the coherent fabric. See 2.5.2 [CPU Core Power Management].

When D18F4x15C[BoostLock]=1, MSRC001_00[6B:64][CpuVid, CpuDid, CpuFid] have special write requirements associated with them.

Table 263: P-state Definitions

Term	Definition
CoreCOF	Core current operating frequency in MHz. CoreCOF = 100 * (MSRC001_00[6B:64][CpuFid[5:0]] + 10h) / (2^MSRC001_00[6B:64][CpuDid]).

Bits	Description
63	**PstateEn**. Read-write. 1=The P-state specified by this MSR is valid. 0=The P-state specified by this MSR is not valid. The purpose of this register is to indicate if the rest of the P-state information in the register is valid after a reset; it controls no hardware.
62:42	RAZ.

41:40	**IddDiv: current divisor**. Read-write. See IddValue.
39:32	**IddValue: current value**. Read-write. After a reset, IddDiv and IddValue combine to specify the expected maximum current dissipation of a single core that is in the P-state corresponding to the MSR number. These values are intended to be used to create ACPI-defined _PSS objects (see 2.5.2.1.7.3 [ACPI Processor P-state Objects]) and to perform the 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check]. The values are expressed in amps; they are not intended to convey final product power levels; they may not match the power levels specified in the Power and Thermal Datasheets. These fields are encoded as follows: IddDiv — Description 00b — IddValue / 1 A, Range: 0 to 255 A. 01b — IddValue / 10 A, Range: 0 to 25.5 A. 10b — IddValue / 100 A, Range: 0 to 2.55 A. 11b — Reserved
31:23	RAZ.
22	**NbPstate: Northbridge P-state**. IF (MSRC001_0071[NbPstateDis]) THEN Read-only. ELSE Read-write. ENDIF. 1=Low performance NB P-state. 0=High performance NB P-state. If this bit is set in any given P-state register, then it must also be set in all enabled lower performance P-state registers as well. Equivalent P-states in each core must program this bit to the same value. See 2.5.3.1 [NB P-states] and D18F5x170[NbPstateThreshold, NbPstateLo, NbPstateHi].
21	RAZ.
20:17	RAZ.
16	**CpuVid[7]: core VID bit[7]**. Read-write. Except as required by 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check], software should not modify this field. See CpuVid[6:0].
15:9	**CpuVid[6:0]: core VID**. Read-write. Except as required by 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check], software should not modify this field. See 2.5.1 [Processor Power Planes And Voltage Control].
8:6	**CpuDid: core divisor ID**. Read-write. Except as required by 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check], software should not modify this field. Specifies the core frequency divisor; see CpuFid. Bits — Description 0h — Divide by 1 1h — Divide by 2 2h — Divide by 4 3h — Divide by 8 4h — Divide by 16 7h-5h — Reserved.
5:0	**CpuFid[5:0]: core frequency ID**. Read-write. Except as required by 2.5.2.1.6 [Processor-Systemboard Current Delivery Compatibility Check], software should not modify this field. Specifies the core frequency multiplier. The core COF is a function of CpuFid and CpuDid, and defined by CoreCOF.

MSRC001_0070 COFVID Control

Cold reset: Product-specific. There is one register implemented for each core. This register includes several fields that are identical to MSRC001_00[6B:64]. It is controlled by hardware for P-state transitions. It may also be used by software to directly control the current COF or VID.
Accesses to this register that result in invalid COFs or VIDs are ignored. See 2.5.2 [CPU Core Power Management].

Bits	Description
63:32	RAZ.
31:0	Alias of D18F3xC0.

MSRC001_0071 COFVID Status

See 2.5.2 [CPU Core Power Management].

Bits	Description
63:32	Alias of D18F3xCC.
31:0	Alias of D18F3xC8.

MSRC001_0073 C-state Base Address

Reset: 0000_0000_0000_0000h.

Bits	Description
63:16	Reserved.
15:0	**CstateAddr: C-state address**. Read-write. Specifies the IO addresses trapped by the core for C-state entry requests. A value of 0 in this field specifies that the core does not trap any IO addresses for C-state entry. Writing values greater than FFF8h into this field result in undefined behavior. All other values cause the core to trap IO addresses CstateAddr through CstateAddr+7. See 2.5.2.2.2 [C-state Request Interface], D18F4x11[C:8], and D18F4x11C.

MSRC001_0074 CPU Watchdog Timer (CpuWdtCfg)

Read-write; Same-for-all. Reset: 0000_0000_0000_0000h.

The CPU watchdog timer (WDT) is implemented as a counter that counts out the time periods specified. The counter starts counting when CpuWdtEn is set. The counter does not count during halt or stop-grant. It restarts the count each time an operation of an instruction completes. If no operation completes by the specified time period, then a machine check error may be recorded if enabled (see MSR0000_0414 through MSR0000_0417). If a watchdog timer error overflow occurs (MSR0000_0415[Overflow]), a Sync flood can be generated if enabled in D18F3x180[SyncFloodOnCpuLeakErr].

The CPU watchdog timer must be set higher than the NB watchdog timer (D18F3x44 [MCA NB Configuration]) in order to allow remote requests to complete. The CPU watchdog timer must be set the same for all CPUs in a system.

Bits	Description
63:7	Reserved.

<table>
<tr><td>6:3</td><td>CpuWdtCountSel: CPU watchdog timer count select. BIOS: 1. CpuWdtCountSel and CpuWdtTimeBase together specify the time period required for the WDT to expire. The time period is ((the multiplier specified by CpuWdtCountSel) * (the time base specified by CpuWdtTimeBase)). The actual timeout period may be anywhere from zero to one increments less than the values specified, due to non-deterministic behavior.
<table>
<tr><th>Bits</th><th>Multiplier</th><th>Bits</th><th>Multiplier</th></tr>
<tr><td>0h</td><td>4095</td><td>6h</td><td>63</td></tr>
<tr><td>1h</td><td>2047</td><td>7h</td><td>31</td></tr>
<tr><td>2h</td><td>1023</td><td>8h</td><td>8191</td></tr>
<tr><td>3h</td><td>511</td><td>9h</td><td>16383</td></tr>
<tr><td>4h</td><td>255</td><td>Fh-Ah</td><td>Reserved</td></tr>
<tr><td>5h</td><td>127</td><td></td><td></td></tr>
</table></td></tr>
<tr><td>2:1</td><td>CpuWdtTimeBase: CPU watchdog timer time base. Specifies the time base for the timeout period specified in CpuWdtCountSel.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>00b</td><td>1.31 ms</td></tr>
<tr><td>01b</td><td>1.28 us</td></tr>
<tr><td>10b</td><td>Reserved</td></tr>
<tr><td>11b</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>0</td><td>CpuWdtEn: CPU watchdog timer enable. BIOS: 1. 1=The WDT is enabled.</td></tr>
</table>

MSRC001_007A Compute Unit Power Accumulator

Read-only; Updated-by-hardware; Per-compute-unit. Reset: 0.

Bits	Description
63:0	**CmpUnitPwrAcc**. Specifies the total accumulated power in a compute unit. This field rolls over and does not saturate.

MSRC001_007B Max Compute Unit Power Accumulator

Read-only; Updated-by-hardware; Per-compute-unit. Reset: 0.

Bits	Description
63:0	**MaxCmpUnitPwrAcc**. Specifies the maximum accumulated power in a compute unit.

MSRC001_0111 SMM Base Address (SMM_BASE)

Reset: 0000_0000_0003_0000h. This holds the base of the SMM memory region. The value of this register is stored in the save state on entry into SMM (see 2.4.9.2.5 [SMM Save State]) and it is restored on returning from SMM. The 16-bit CS (code segment) selector is loaded with SmmBase[19:4] on entering SMM. SmmBase[3:0] is required to be 0. The SMM base address can be changed in two ways:

- The SMM base address, at offset FF00h in the SMM state save area, may be changed by the SMI handler. The RSM instruction updates SmmBase with the new value.
- Normal WRMSR access to this register.

Bits	Description
63:32	Reserved.
31:0	**SmmBase**. IF MSRC001_0015[SmmLock] THEN Read-only; Not-same-for-all. ELSE Read-write; Not-same-for-all. ENDIF.

MSRC001_0112 SMM TSeg Base Address (SMMAddr)

Reset: 0000_0000_0000_0000h.

See 2.4.9.2 [System Management Mode (SMM)] and 2.4.6.1 [Memory Access to the Physical Address Space]. See MSRC001_0113 for more information about the ASeg and TSeg address ranges.

Each CPU access, directed at CPUAddr, is determined to be in the TSeg range if the following is true:

CPUAddr[47:17] & TSegMask[47:17] == TSegBase[47:17] & TSegMask[47:17].

For example, if TSeg spans 256 KB and starts at the 1 MB address. The MSRC001_0112[TSegBase] would be set to 0010_0000h and the MSRC001_0113[TSegMask] to FFFC_0000h (with zeros filling in for bits[16:0]). This results in a TSeg range from 0010_0000 to 0013_FFFFh.

Bits	Description
63:48	Reserved.
47:17	**TSegBase[47:17]: TSeg address range base**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF.
16:0	Reserved.

MSRC001_0113 SMM TSeg Mask (SMMMask)

Reset: 0000_0000_0000_0000h. See 2.4.9.2 [System Management Mode (SMM)].
The ASeg address range is located at a fixed address from A0000h–BFFFFh. The TSeg range is located at a variable base (specified by MSRC001_0112[TSegBase]) with a variable size (specified by MSRC001_0113[TSegMask]). These ranges provide a safe location for SMM code and data that is not readily accessible by non-SMM applications. The SMI handler can be located in one of these two ranges, or it can be located outside these ranges. These ranges must never overlap each other.

This register specifies how accesses to the ASeg and TSeg address ranges are controlled as follows:

- If [A, T]Valid == 1, then:
 - If in SMM, then:
 - If [A, T]Close == 0, then the accesses are directed to DRAM with memory type as specified in [A, T]MTypeDram.
 - If [A, T]Close == 1, then instruction accesses are directed to DRAM with memory type as specified in [A, T]MTypeDram and data accesses are directed at MMIO space and with attributes based on [A, T]MTypeIoWc.
 - If not in SMM, then the accesses are directed at MMIO space with attributes based on [A, T]MTypeIoWc.
- See 2.4.6.1.1 [Determining Memory Type].

Bits	Description
63:48	Reserved.
47:17	**TSegMask[47:17]: TSeg address range mask**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. See MSRC001_0112.
16:15	Reserved.
14:12	**TMTypeDram: TSeg address range memory type**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. Specifies the memory type for SMM accesses to the TSeg range that are directed to DRAM. See: Table 235 [Valid Values for Memory Type Definition].
11	Reserved.
10:8	**AMTypeDram: ASeg Range Memory Type**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. Specifies the memory type for SMM accesses to the ASeg range that are directed to DRAM. See: Table 235 [Valid Values for Memory Type Definition].
7:6	Reserved.
5	**TMTypeIoWc: non-SMM TSeg address range memory type**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. Specifies the attribute of TSeg accesses that are directed to MMIO space. 0=UC (uncacheable). 1=WC (write combining).
4	**AMTypeIoWc: non-SMM ASeg address range memory type**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. Specifies the attribute of ASeg accesses that are directed to MMIO space. 0=UC (uncacheable). 1=WC (write combining).
3	**TClose: send TSeg address range data accesses to MMIO**. Read-write. 1=When in SMM, direct data accesses in the TSeg address range to MMIO space. See AClose.
2	**AClose: send ASeg address range data accesses to MMIO**. Read-write. 1=When in SMM, direct data accesses in the ASeg address range to MMIO space. [A, T]Close allows the SMI handler to access the MMIO space located in the same address region as the [A, T]Seg. When the SMI handler is finished accessing the MMIO space, it must clear the bit. Failure to do so before resuming from SMM causes the CPU to erroneously read the save state from MMIO space.
1	**TValid: enable TSeg SMM address range**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. 1=The TSeg address range SMM enabled.
0	**AValid: enable ASeg SMM address range**. IF MSRC001_0015[SmmLock] THEN Read-only. ELSE Read-write. ENDIF. 1=The ASeg address range SMM enabled.

MSRC001_0114 Virtual Machine Control (VM_CR)

Bits	Description
63:32	Reserved.
31:5	MBZ.
4	**SvmeDisable: SVME disable**. See Lock for the access type of this field. Reset: 0. 1=MSRC000_0080[SVME] is MBZ. 0=MSRC000_0080[SVME] is Read-write. Attempting to set this field when (MSRC000_0080[SVME] == 1) causes a #GP fault, regardless of the state of Lock. See the APM2 section titled “Enabling SVM” for software use of this field.

3	**Lock: SVM lock**. Read-only; Write-1-only; Cleared-by-hardware. Reset: 0. See MSRC001_0118[SvmLockKey] for the condition that causes hardware to clear this field. 1=SvmeDisable is read-only. 0=SvmeDisable is Read-write.
2	**DisA20m: disable A20 masking**. Read-write; Set-by-hardware. Reset: 0. 1=Disables A20 masking. This bit is set by hardware when the SKINIT instruction is executed.
1	**InterceptInit: intercept INIT**. Read-write; Set-by-hardware. Reset: 0. This bit controls how INIT is delivered in host mode. This bit is set by hardware when the SKINIT instruction is executed. 0=INIT delivered normally. 1=INIT translated into a SX interrupt.
0	**DPD: debug port disable**. Read-write; Set-by-hardware. Reset: 0. Set-by-hardware when the SKINIT instruction is executed. This bit controls if debug facilities such as JTAG and HDT have access to the processor state information. 1=HDT is disabled. 0=HDT may be enabled.

MSRC001_0115 IGNNE

Bits	Description
63:32	Reserved.
31:1	MBZ.
0	**IGNNE: current IGNNE state**. Read-write. Reset: 0. This bit controls the current state of the processor internal IGNNE signal.

MSRC001_0116 SMM Control (SMM_CTL)

IF (MSRC001_0015[SmmLock]) THEN GP-read-write. ELSE GP-read; Write-only. ENDIF.
The bits in this register are processed in the order of: SmmEnter, SmiCycle, SmmDismiss, RsmCycle and SmmExit. However, only the following combination of bits may be set in a single write (all other combinations result in undefined behavior):

- SmmEnter and SmiCycle.
- SmmEnter and SmmDismiss.
- SmmEnter, SmiCycle and SmmDismiss.
- SmmExit and RsmCycle.

Software is responsible for ensuring that SmmEnter and SmmExit operations are properly matched and are not nested.

Bits	Description
63:5	MBZ.
4	**RsmCycle: send RSM special cycle**. 1=Send a RSM special cycle.
3	**SmmExit: exit SMM**. 1=Exit SMM.
2	**SmiCycle: send SMI special cycle**. 1=Send a SMI special cycle.
1	**SmmEnter: enter SMM**. 1=Enter SMM.
0	**SmmDismiss: clear SMI**. 1=Clear the SMI pending flag.

MSRC001_0117 Virtual Machine Host Save Physical Address (VM_HSAVE_PA)

Bits	Description
63:48	MBZ.
47:12	**VM_HSAVE_PA: physical address of host save area**. Read-write. Reset: 0. This register contains the physical address of a 4-KB region where VMRUN saves host state and where vm-exit restores host state from. Writing this register causes a #GP if (FF_FFFF_Fh >= VM_HSAVE_PA >= FD_0000_0h).
11:0	MBZ.

MSRC001_0118 SVM Lock Key

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**SvmLockKey: SVM lock key**. RAZ; Write. Writes to this register when MSRC001_0114[Lock] == 0 modify SvmLockKey. If ((MSRC001_0114[Lock] == 1) && (SvmLockKey != 0) && (The write value == The value stored in SvmLockKey)) for a write to this register then hardware updates MSRC001_0114[Lock] = 0.

MSRC001_011A Local SMI Status

Reset: 0000_0000_0000_0000h. This registers returns the same information that is returned in SMMFEC4 [Local SMI Status] portion of the SMM save state. The information in this register is only updated when MSRC001_0116[SmmDismiss] is set by software.

Bits	Description
63:32	Reserved.
31:0	Alias of SMMFEC4 [Local SMI Status].

MSRC001_011B AVIC Doorbell

GP-read; Write-only. Reset: 0. Support for this MSR is indicated by (CPUID Fn8000_000A_EDX[AVIC] == 1). The ApicId is a physical APIC Id; not valid for logical APIC ID.

Bits	Description
63:8	MBZ.
7:0	**ApicId: APIC ID[7:0]**.

MSRC001_0140 OS Visible Work-around MSR0 (OSVW_ID_Length)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:16	Reserved.
15:0	**OSVWIdLength: OS visible work-around ID length**. Read-write. See the Revision Guide for the definition of this field; see 1.2 [Reference Documents].

MSRC001_0141 OS Visible Work-around MSR1 (OSVW Status)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**OsvwStatusBits: OS visible work-around status bits**. Read-write. See the Revision Guide for the definition of this field; see 1.2 [Reference Documents].

MSRC001_020[A,8,6,4,2,0] Performance Event Select (PERF_CTL[5:0])

Reset: 0000_0000_0000_0000h. See 2.6.1 [Performance Monitor Counters]. MSRC001_00[03:00] is an alias of MSRC001_020[6,4,2,0].

Table 264: Register Mapping for MSRC001_020[A,8,6,4,2,0]

Register	Function
MSRC001_0200	Counter 0
MSRC001_0202	Counter 1
MSRC001_0204	Counter 2
MSRC001_0206	Counter 3
MSRC001_0208	Counter 4
MSRC001_020A	Counter 5

Bits	Description
63:42	Reserved.
41:40	**HostGuestOnly: count only host/guest events**. Read-write. Bits — Description 00b — Count all events, irrespective of guest/host. 01b — Count guest events if MSRC000_0080[SVME] == 1. 10b — Count host events if MSRC000_0080[SVME] == 1. 11b — Count all guest and host events if MSRC000_0080[SVME] == 1.
39:36	Reserved.
35:32	**EventSelect[11:8]: performance event select**. See: EventSelect[7:0].

31:24	**CntMask: counter mask**. Read-write. Controls the number of events counted per clock cycle. Bits — Description 00h — The corresponding PERF_CTR[5:0] register increments by the number of events occurring in a clock cycle. Maximum number of events in one cycle is 32. 7Fh-01h — When Inv == 0, the corresponding PERF_CTR[5:0] register increments by 1, if the number of events occurring in a clock cycle is greater than or equal to the CntMask value. When Inv == 1, the corresponding PERF_CTR[5:0] register increments by 1, if the number of events occurring in a clock cycle is less than CntMask value. FFh-80h — Reserved.
23	**Inv: invert counter mask**. Read-write. See CntMask.
22	**En: enable performance counter**. Read-write. 1= Performance event counter is enabled.
21	Reserved.
20	**Int: enable APIC interrupt**. Read-write. 1=APIC performance counter LVT interrupt is enabled to generate an interrupt via APIC340 [LVT Performance Monitor] when the performance counter overflows.
19	Reserved.
18	**Edge: edge detect**. Read-write. 0=Level detect. 1=Edge detect. The edge count mode increments the counter when a transition happens on the monitored event. If the event selected is changed without disabling the counter, an extra edge is falsely detected when the first event is a static 0 and the second event is a static one. To avoid this false edge detection, disable the counter when changing the event and then enable the counter with a second MSR write.
17:16	**OsUserMode: OS and user mode**. Read-write. Bits — Description 00b — Count no events. 01b — Count user events (CPL > 0). 10b — Count OS events (CPL == 0). 11b — Count all events, irrespective of the CPL.
15:8	**UnitMask: event qualification**. Read-write. Each UnitMask bit further specifies or qualifies the event specified by EventSelect. All events selected by UnitMask are simultaneously monitored. Unless otherwise stated, the UnitMask values shown may be combined (logically ORed) to select any desired combination of the sub-events for a given event. In some cases, certain combinations can result in misleading counts, or the UnitMask value is an ordinal rather than a bit mask. These situations are described where applicable, or should be obvious from the event descriptions. For events where no UnitMask table is shown, the UnitMask is Unused.
7:0	**EventSelect[7:0]: event select**. Read-write. EventSelect[11:0] = {EventSelect[11:8], EventSelect[7:0]}. EventSelect specifies the event or event duration in a processor unit to be counted by the corresponding PERF_CTR[5:0] register. The events are specified in 3.24 [Core Performance Counter Events]. Some events are reserved; when a reserved event is selected, the results are undefined.

MSRC001_020[B,9,7,5,3,1] Performance Event Counter (PERF_CTR[5:0])

See MSRC001_020[A,8,6,4,2,0] [Performance Event Select (PERF_CTL[5:0])]. MSRC001_00[07:04] is an alias of MSRC001_020[7,5,3,1].

Table 265: Register Mapping for MSRC001_020[B,9,7,5,3,1]

Register	Function
MSRC001_0201	Counter 0
MSRC001_0203	Counter 1
MSRC001_0205	Counter 2
MSRC001_0207	Counter 3
MSRC001_0209	Counter 4
MSRC001_020B	Counter 5

Bits	Description
63:48	RAZ.
47:0	**CTR: performance counter value**. Read-write. Reset: 0.

MSRC001_024[6,4,2,0] Northbridge Performance Event Select (NB_PERF_CTL[3:0])

Per-node. See 2.6.1 [Performance Monitor Counters]. MSRC001_024[6,4,2,0][31:0] is an alias of D18F5x[70,60,50,40]. MSRC001_024[6,4,2,0][63:32] is an alias of D18F5x[74,64,54,44].

Table 266: Register Mapping for MSRC001_024[6,4,2,0]

Register	Function
MSRC001_0240	Counter 0
MSRC001_0242	Counter 1
MSRC001_0244	Counter 2
MSRC001_0246	Counter 3

Note: To get meaningful data, each of the counters should be similiarly programmed across events selected.

Bits	Description
63:36	Reserved.
35:32	**EventSelect[11:8]: performance event select**. Read-write. Reset: 0. See EventSelect[7:0].
31:23	Reserved.
22	**En: enable performance counter**. Read-write. Reset: 0. 1=Performance event counter is enabled.
21	Reserved.
20	**Int: enable APIC interrupt**. Read-write. Reset: 0. 1=APIC performance counter LVT interrupt is enabled to generate an interrupt via APIC340 [LVT Performance Monitor] to all local APIC's on this node when the performance counter overflows.
19:16	Reserved.

15:8	**UnitMask: event qualification**. Read-write. Reset: 0. Each UnitMask bit further specifies or qualifies the event specified by EventSelect. All events selected by UnitMask are simultaneously monitored. Unless otherwise stated, the UnitMask values shown may be combined (logically ORed) to select any desired combination of the sub-events for a given event. In some cases, certain combinations can result in misleading counts, or the UnitMask value is an ordinal rather than a bit mask. These situations are described where applicable, or should be obvious from the event descriptions. For events where no UnitMask table is shown, the UnitMask is Unused.
7:0	**EventSelect[7:0]: event select**. Read-write. Reset: 0. This field, along with EventSelect[11:8] above, combine to form the 12-bit event select field, EventSelect[11:0]. EventSelect specifies the event or event duration in a processor unit to be counted by the corresponding NB_PERF_CTR[3:0] register. The events are specified in 3.25 [NB Performance Counter Events]. Some events are reserved; when a reserved event is selected, the results are undefined.

MSRC001_024[7,5,3,1] Northbridge Performance Event Counter (NB_PERF_CTR[3:0])

Per-node. See MSRC001_024[6,4,2,0] [Northbridge Performance Event Select (NB_PERF_CTL[3:0])]. MSRC001_024[7,5,3,1][31:0] is an alias of D18F5x[78,68,58,48]. MSRC001_024[7,5,3,1][63:32] is an alias of D18F5x[7C,6C,5C,4C].
See 2.6.1.2 [NB Performance Monitor Counters] for proper read sequence.

Table 267: Register Mapping for MSRC001_024[7,5,3,1]

Register	Function
MSRC001_0241	Counter 0
MSRC001_0243	Counter 1
MSRC001_0245	Counter 2
MSRC001_0247	Counter 3

Bits	Description
63:48	RAZ.
47:32	**CTR[47:32]: performance counter value[47:32]**. See: CTR[31:0].
31:0	**CTR[31:0]: performance counter value[31:0]**. Read-write. Reset: 0. CTR[47:0] = {CTR[47:32], CTR[31:0]}. Returns the current value of the event counter.

MSRC001_0280 Performance Time Stamp Counter (CU_PTSC)

Support for MSRC001_0280 [Performance Time Stamp Counter (CU_PTSC)] indicated by CPUID Fn8000_0001_ECX[PerfTsc]. The size of PTSC indicated by CPUID Fn8000_0008_ECX[PerfTscSize]. Increments at a 100 MHz rate in all P-states, all C states, S0, or S1. Each core on a node at the same instant in time varies by +/- 3 100 MHz clocks. The value of PTSC[31:0] is inserted into each record produced.

Bits	Description
63:40	RAZ.
39:0	**PTSC: global timestamp counter**. Read-only; Updated-by-hardware. Reset: 0.

3.23 MSRs - MSRC001_1xxx

MSRC001_1002 CPUID Features for CPUID Fn0000_0007_E[B,A]X_x0

Read-write. Reset: {CPUID Fn0000_0007_EAX_x0, CPUID Fn0000_0007_EBX_x0}. MSRC001_1002[63:32] provides control over values read from CPUID Fn0000_0007_EAX_x0; MSRC001_1002[31:0] provides control over values read from CPUID Fn0000_0007_EBX_x0.

Bits	Description
63:9	Reserved.
8	**BMI2**.
7	**SMEP**.
6	Reserved.
5	**AVX2**.
4	Reserved.
3	**BMI1**.
2:1	Reserved.
0	**FSGSBASE**.

MSRC001_1003 Thermal and Power Management CPUID Features

MSRC001_1003 provides control over values read from CPUID Fn0000_0006_ECX.

Bits	Description
63:32	Reserved.
31:0	**FeaturesEcx**. Read-write. Reset: CPUID Fn0000_0006_ECX. Provides control over the features reported in CPUID Fn0000_0006_ECX.

MSRC001_1004 CPUID Features (Features)

Read-write. Reset: {CPUID Fn0000_0001_ECX, CPUID Fn0000_0001_EDX}. MSRC001_1004[63:32] provides control over values read from CPUID Fn0000_0001_ECX; MSRC001_1004[31:0] provides control over values read from CPUID Fn0000_0001_EDX.

Bits	Description
63	Reserved.
62	**RDRAND**.
61	**F16C**.
60	**AVX**.
59	**OSXSAVE**. Modifies CPUID Fn0000_0001_ECX[OSXSAVE] only if CR4[OSXSAVE].
58	**XSAVE**.
57	**AES**. Modifies CPUID Fn0000_0001_ECX[AES] only if the reset value is 1.
56	Reserved.
55	**POPCNT**.

54	**MOVBE**.
53	**x2APIC**.
52	**SSE42**.
51	**SSE41**.
50:46	Reserved.
45	**CMPXCHG16B**.
44:42	Reserved.
41	**SSSE3**.
40:36	Reserved.
35	**Monitor**. Modifies CPUID Fn0000_0001_ECX[Monitor] only if ~MSRC001_0015[MonMwaitDis].
34	Reserved.
33	**PCLMULQDQ**. Modifies CPUID Fn0000_0001_ECX[PCLMULQDQ] only if the reset value is 1.
32	**SSE3**.
31:29	Reserved.
28	**HTT**.
27	Reserved.
26	**SSE2**.
25	**SSE**.
24	**FXSR**.
23	**MMX™**.
22:20	Reserved.
19	**CLFSH**.
18	Reserved.
17	**PSE36**.
16	**PAT**.
15	**CMOV**.
14	**MCA**.
13	**PGE**.
12	**MTRR**.
11	**SysEnterSysExit**.
10	Reserved.
9	**APIC**. Modifies CPUID Fn0000_0001_EDX[APIC] only if MSR0000_001B[ApicEn].
8	**CMPXCHG8B**.

7	**MCE**.
6	**PAE**.
5	**MSR**.
4	**TSC**.
3	**PSE**.
2	**DE**.
1	**VME**.
0	**FPU**.

MSRC001_1005 Extended CPUID Features (ExtFeatures)

Read-write. Reset: {CPUID Fn8000_0001_ECX, CPUID Fn8000_0001_EDX}. MSRC001_1005[63:32] provides control over values read from CPUID Fn8000_0001_ECX; MSRC001_1005[31:0] provides control over values read from CPUID Fn8000_0001_EDX.

Bits	Description
63:61	Reserved.
60	**PerfCtrExtL2I**.
59	**PerfTsc**.
58	**DataBreakpointExtension**.
57	Reserved.
56	**PerfCtrExtNB**.
55	**PerfCtrExtCore**.
54	**TopologyExtensions**.BIOS: 0.
53	**TBM**.
52	Reserved.
51	**NodeId**. BIOS: 1.
50	Reserved.
49	Reserved.
48	**FMA4**.
47	**LWP**.
46	Reserved.
45	**WDT**.
44	**SKINIT**.
43	**XOP**.
42	**IBS**.
41	**OSVW**.
40	**3DNowPrefetch**.
39	**MisAlignSse**.
38	**SSE4A**.
37	**ABM**.

36	**AltMovCr8**.
35	**ExtApicSpace**.
34	**SVM**. Modifies CPUID Fn8000_0001_ECX[SVM] only if D18F3xE8[SvmCapable].
33	**CmpLegacy**.
32	**LahfSahf**.
31	**3DNow**.
30	**3DNowExt**.
29	**LM**. Read-write.
28	Reserved.
27	**RDTSCP**.
26	**Page1GB**.
25	**FFXSR**.
24	**FXSR**.
23	**MMX™**.
22	**MmxExt**.
21	Reserved.
20	**NX**.
19:18	Reserved.
17	**PSE36**.
16	**PAT**.
15	**CMOV**.
14	**MCA**.
13	**PGE**.
12	**MTRR**.
11	**SysCallSysRet**.
10	Reserved.
9	**APIC**.
8	**CMPXCHG8B**.
7	**MCE**.
6	**PAE**.
5	**MSR**.
4	**TSC**.
3	**PSE**.
2	**DE**.
1	**VME**.
0	**FPU**.

MSRC001_101[B:9] Address Mask For DR[3:1] Breakpoints

Reset: 0000_0000_0000_0000h. Support indicated by CPUID Fn8000_0001_ECX[DataBreakpointExtension]. See MSRC001_1027.

Table 268: Register Mapping for MSRC001_101[B:9]

Register	Function
MSRC001_1019	DR1_ADDR_MASK
MSRC001_101A	DR2_ADDR_MASK
MSRC001_101B	DR3_ADDR_MASK

Table 269: Field Mapping for MSRC001_101[B:9]

Register	Bits
	31:0
MSRC001_1019	DR1
MSRC001_101A	DR2
MSRC001_101B	DR3

Bits	Description
63:32	Reserved.
31:0	**AddrMask: mask for DR linear address data breakpoint**. Read-write. This field qualifies the DR linear address data breakpoint, allowing the DR[3:1] data breakpoint on a range of addresses in memory. The mask bits are active high; 0=Include bit into address compare; 1=Exclude bit into address compare. AddrMask is always used, and it can be used in conjunction with any debug function that uses DR[3:1]. The legacy DR breakpoint function is provided by AddrMask[31:0] == 0000_0000h).

MSRC001_101C Load-Store Configuration 3 (LS_CFG3)

Bits	Description
63:26	Reserved.
25	**DisPfHwForSw**. Read-write. Reset: 0. 1=Disable hardware prefetches for software prefetches.
24	Reserved.
23	**DisHwPf**. Read-write. Reset: 0. 1=Disable the DC hardware prefetcher. BIOS: See 2.3.3 [Using L2 Cache as General Storage During Boot].
22:21	Reserved.
20	**DisSpecTlbRld**. Read-write. Reset: 0. 1=Disable speculative TLB reloads. BIOS: See 2.3.3 [Using L2 Cache as General Storage During Boot].
19:8	Reserved.
7	**DisSS**. Read-write; Same-for-all. Reset: 0. BIOS: See 2.3.3 [Using L2 Cache as General Storage During Boot]. 1=Disable streaming store functionality.
6:0	Reserved.

MSRC001_1020 Load-Store Configuration (LS_CFG)

Bits	Description
63:0	Reserved.

MSRC001_1021 Instruction Cache Configuration (IC_CFG)

Per-compute-unit.

Bits	Description
63:10	Reserved.
9	**DisSpecTlbRld**. Read-write. Reset: 0. 1=Disable speculative IC TLB reload request; the request is not made to the TLB walker until the fetch is non-speculative. BIOS: See 2.3.3 [Using L2 Cache as General Storage During Boot].
8:0	Reserved.

MSRC001_1022 Data Cache Configuration (DC_CFG)

Bits	Description
63:0	Reserved.

MSRC001_1023 Combined Unit Configuration (CU_CFG)

Per-compute-unit.

Bits	Description
63:52	Reserved.
51	**EarlyNbDataReturnEn**. Read-write. Reset: 0. BIOS: 1. 1=Enable faster data return from NB to L1 caches if NB returns an entire block of data in contiguous cycles.
50	Reserved.
49	**ProcFeedbackEn: processor feedback interface enable**. Read-write. Reset: 0. BIOS: 1. 1=Enable processor feedback interface; CPUID Fn8000_0007_EAX.
48:24	Reserved.
23	**L2WayLock: L2 way lock enable**. Read-write. Reset: 0. 1=Allocations and evictions for the L2 ways >= L2FirstLockedWay are disabled. Probes can still invalidate a line in a locked way. Cache lines in the locked ways of the L2 are still accessible by software. See 2.3.3 [Using L2 Cache as General Storage During Boot].
22:19	**L2FirstLockedWay: first L2 way locked**. Read-write. Reset: 0h. See L2WayLock. Bits — Description 0h — Reserved. Eh-1h — Ways <L2FirstLockedWay> to 15 locked. Fh — Way 15 locked.
18:0	Reserved.

MSRC001_1027 Address Mask For DR0 Breakpoints (DR0_ADDR_MASK)

Reset: 0000_0000_0000_0000h. Support for AddrMaskDR0[31:12] is indicated by CPUID Fn8000_0001_ECX[DataBreakpointExtension]. See MSRC001_101[B:9].

Bits	Description
63:32	Reserved.
31:0	**AddrMaskDR0: mask for DR0 linear address data breakpoint**. Read-write. This field qualifies the DR0 linear address data breakpoint, allowing the DR0 data breakpoint on a range of addresses in memory. The mask bits are active high; 0=Include bit into address compare; 1=Exclude bit into address compare. AddrMaskDR0 is always used, and it can be used in conjunction with any debug function that uses DR0. AddrMaskDR0[31:12] is only valid for data breakpoints. The legacy DR0 breakpoint function is provided by AddrMaskDR0[31:0] == 0000_0000h).

MSRC001_1028 Floating Point Configuration (FP_CFG)

Per-compute-unit.

Bits	Description
63:45	Reserved.
44:41	**DiDtCfg4**. Read-write. Reset: 1111b. BIOS: D0F0xBC_xD021_1074[DiDtCfg4].
40	**DiDtCfg3**. Read-write. Reset: 0. BIOS: D0F0xBC_xD021_1074[DiDtCfg3].
39:35	Reserved.
34:27	**DiDtCfg1**. Read-write. Reset: 1001_1011b. BIOS: D0F0xBC_xD021_1074[DiDtCfg1].
26:25	**DiDtCfg2**. Read-write. Reset: 00b. BIOS: D0F0xBC_xD021_1074[DiDtCfg2].
24:23	Reserved.
22:18	**DiDtCfg0**. Read-write. Reset: 1_1111b. BIOS: D0F0xBC_xD021_1074[DiDtCfg0].
17	Reserved.
16	**DiDtMode**. Read-write. Reset: 0. BIOS: D0F0xBC_xD021_1074[DiDtMode].
15:0	Reserved.

MSRC001_102A Combined Unit Configuration 2 (CU_CFG2)

Per-compute-unit.

Bits	Description
63:57	Reserved.
56:52	**L2UpsizeCUCT: L2 upsize detector Commited Micro-op Counter Threshold**. Read-write. Reset: 0.
51	Reserved.

50	**RdMmExtCfgQwEn: read mmio extended config quadword enable**. Read-write. Reset: 0. BIOS: 1. 1=MMIO reads to extended config space do not need to be doubleword aligned and may be up to quadword sized. This is to support 64-bit MMIO reads to extended config space. 0=MMIO reads to extended config space need to be doubleword aligned and may be up to doubleword sized. MMIO reads to extended config space that are either not doubleword aligned or greater than doubleword sized are treated as plain MMIO reads.
49:38	Reserved.
37:36	**ThrottleNbInterface[3:2]**. Read-write. Reset: 01b. BIOS: 00b. See ThrottleNbInterface[1:0].
35:22	Reserved.
21:18	**L2UpsizeCSWT[4:1]: L2 upsize context switch warmup threshold [4:1]**. Read-write. Reset: 0. BIOS. 0000b. See L2UpsizeCSWT[0].
17:11	Reserved.
10	**VicResyncChkEn**. Read-write. Reset: 0. BIOS: 1. 1=Generate an internal probe to NB for non-shared victims. Required to be set for the Monitor/MWait instructions.
9	**L2UpsizeCSWT[0]: L2 upsize context switch warmup threshold [0]**. Read-write. Reset: 0. BIOS. 0b. L2UpsizeCSWT[4:0] = {L2UpsizeCSWT[4:1], L2UpsizeCSWT[0]}. The L2UpsizeCSWT defines the context switch warmup threshold for L2 cache upsizing. When non-zero, both the internal eviction count and the internal committed uop count are cleared when either a context switch occurs (indicated by MOV CR3) or the committed uop count exceeds the (L2UpsizeCSWT[4:0] * 8K) following a context switch. This defines a period where many rapid evictions can occur without causing an L2 cache upsize following a context switch.
8	Reserved.
7:6	**ThrottleNbInterface[1:0]**. Read-write. ThrottleNbInterface[3:0] = {ThrottleNbInterface[3:2], ThrottleNbInterface[1:0]}. Reset: 11b. BIOS: NumOfCompUnits-1. Specifies how many clocks the CU needs to wait before sending the next packet of information to the NB. This applies to the CU->NB request interface and the CU->NB probe response interface. This field must be programmed to a value greater than or equal to the number of compute units in the node that have at least one enabled core minus 1. See 2.4.4 [Processor Cores and Downcoring]. Bits — Description 0h — 0 Clocks. 1h — 1 Clock. 2h — 2 Clocks. 3h — 3 Clocks. Fh-4h — Reserved.
5	Reserved.
4:0	**L2UpsizeERT: L2 upsize evict rate threshold**. Read-write. Reset: 0. BIOS: 0_0000b. The L2UpsizeERT defines the eviction rate threshold that can cause an L2 cache upsize after a CC6 exit. When L2UpsizeERT is zero, the L2 cache upsize mechanism is disabled. When L2UpsizeERT is non-zero, L2 cache upsizes can be triggered after the internal committed uop counter interval expires (defined by L2UpsizeCUCT) if the internal eviction counter equals or exceeds the (L2UpsizeERT[4:0] * 64). Ratios of the L2UpsizeERT and L2UpsizeCUCT support a range of 0.00024 through 0.25 evictions per committed uop for L2 cache upsize tuning.

MSRC001_102B Combined Unit Configuration 3 (CU_CFG3)

Per-compute-unit.

Bits	Description
63:61	Reserved.
60:59	**PcidDecrScaleFactor**. Read-write. Reset: 00b. Specifies the decrement rate for the PCID replacement counter for PCIDs not currently in use; The larger the value programmed the slower the counter decrements. Bits — Description 00b — Inactive PCID replacement counter decrements every 64 TLB replacements. 01b — Inactive PCID replacement counter decrements every 128 TLB replacements. 10b — Inactive PCID replacement counter decrements every 256 TLB replacements. 11b — Inactive PCID replacement counter decrements every 512 TLB replacements.
58:57	**PcidIncrScaleFactor**. Read-write. Reset: 00b. Specifies the increment rate for the PCID replacement counter for PCIDs currently in use; The larger the value programmed the slower the counter increments. Bits — Description 00b — Active PCID replacement counter increments every 16 TLB inserts. 01b — Active PCID replacement counter increments every 32 TLB inserts. 10b — Active PCID replacement counter increments every 64 TLB inserts. 11b — Active PCID replacement counter increments every 128 TLB inserts.
56:55	**AsidIncrScaleFactor**. Read-write. Reset: 00b. Specifies the increment rate for the ASID replacement counter for ASIDs currently in use; The larger the value programmed the slower the counter increments. Bits — Description 00b — Active ASID replacement counter increments every 16 TLB inserts. 01b — Active ASID replacement counter increments every 32 TLB inserts. 10b — Active ASID replacement counter increments every 64 TLB inserts. 11b — Active ASID replacement counter increments every 128 TLB inserts.
54:53	Reserved.
52:51	**AsidDecrScaleFactor**. Read-write. Reset: 00b. Specifies the decrement rate for the ASID replacement counter for ASIDs not currently in use; The larger the value programmed the slower the counter decrements. Bits — Description 00b — Inactive ASID replacement counter decrements every 64 TLB replacements. 01b — Inactive ASID replacement counter decrements every 128 TLB replacements. 10b — Inactive ASID replacement counter decrements every 256 TLB replacements. 11b — Inactive ASID replacement counter decrements every 512 TLB replacements.
50	Reserved.
49	**CombineCr0Cd: combine CR0[CD] for both cores of a compute unit**. Read-write. Reset: 0. BIOS: 1. BIOS: Must not be set when using L2 cache as general storage during boot; See 2.3.3 [Using L2 Cache as General Storage During Boot]; Must be set before passing control to the OS. 0=The effective CR0[CD] is not affected by the hCR0[CD] on other cores. 1=The effective CR0[CD], for all modes, is forced to 1 if the logical OR of the hCR0[CD] for all other cores on the compute unit is 1. Note that the logical OR does not include the local core hCR0[CD].
48:43	Reserved.

42	**PwcDisableWalkerSharing**. Read-write. Reset: 0. BIOS: 0. 1=Page table walker sharing is disabled. Core 0 uses page walker 0 and Core 1 uses page walker 1.
41:23	Reserved.
22	**PfcDoubleStride**. Read-write. Reset: 0. BIOS: 1. 1=Prefetch N and N+1 offsets ahead of a stride miss instead of just N. N is configurable by PfcStrideMul.
21:20	**PfcStrideMul**. Read-write. Reset: 01b. Specifies the number of stride offsets that are prefetched. Bits Description 00b 3 01b 4 10b 5 11b 6
19	Reserved.
18	**PfcDis**. Read-write. Reset: 0. 1=Prefetcher disabled.
17	**PfcStrideDis**. Read-write. Reset: 0. 1=Stride prefetch generation disabled.
16	**PfcRegionDis**. Read-write. Reset: 0. 1=Region prefetch generation disabled.
15:4	Reserved.
3	**PfcL1TrainDis: stride training to L1 disable**. Read-write. Reset: 0. 1=L1 prefetch training disabled.
2:0	Reserved.

MSRC001_102F Prefetch Throttling Configuration (CU_PFTCFG)

Read-write; Per-compute-unit. Reset: 0000_0000_0000_0000h.
The prefetch throttle mechanism, described as follows, is enabled when (PrefetchThrottlingEn == 1), otherwise all L2 prefetches will be sent to the NB.

A tracking structure is defined that holds 4 NB prefetches. Each of the 4 prefetches in this structure stores a hashed physical address (PAHash[13:0]), computed as (PA[47:34] ^ PA[33:20] ^ PA[19:6]). Every Nth prefetch sent to the NB, where N is specified by CaptureThreshold, will replace the oldest entry in the tracking structure. Each entry in the tracking structure implements a DemandHit indication, that is initialized as 0 when inserted and set to 1 if a demand L2 hit also hits on that entry, where a hit is defined as when the L2 hit PAHash matches the entry PAHash.

Two counters are defined, a current accuracy count (AccCntCurrent[5:0]), warm reset to 0, and a previous accuracy count (AccCntPrevious[5:0]), warm reset to 3Fh. AccCntCurrent is incremented if the entry that is replaced by a tracking structure insertion is DemandHit == 1. After TrackThreshold insertions to the tracking structure, AccCntCurrent is written to AccCntPrevious and AccCntCurrent = 0. All throttling decisions are based on the value of AccCntPrevious.

DCT channel utilization also factors into the decision to throttle prefetches. DCT utilization is indicated by a code called DramBwLevel, ranging from 0 (low utilization) to 2 (high utilization). If DRAM prefetch watermark 2 has been reached then DramBwLevel = 2, else if DRAM prefetch watermark 1 has been reached then DramBwLevel = 1, else DramBwLevel = 0. See D18F2x1B4[DcqBwThrotWm2, DcqBwThrotWm1]. Throttling occurs according to ThrottleLevel as a function of DramBwLevel:

- If ((DramBwLevel == 0) && (AccCntPrevious < AccThresh0)) then throttle.
- If ((DramBwLevel == 1) && (AccCntPrevious < AccThresh1)) then throttle.
- If ((DramBwLevel == 2) && (AccCntPrevious < AccThresh2)) then throttle.

Bits	Description
63:37	Reserved.
36	**PrefetchThrottlingEn: prefetch throttling enable**. BIOS: 0. 1=Prefetch throttling enabled.
35:30	**ThrottleLevel: throttling level**. BIOS: 0. Drop ThrottleLevel prefetches out of every every ThrottleLevel+1 prefetches. Bits — Description 00h — Reserved. 01h — Drop 1 prefetch out of every every 2 prefetches. 3Eh-02h — Drop <ThrottleLevel> prefetches out of every every <ThrottleLevel+1> prefetches. 3Fh — Drop 63 prefetches out of every every 64 prefetches.
29:24	**AccThresh2: accuracy threshold level 2**. BIOS: 0. Throttle prefetches if (DramBwLevel == 2) and (AccCntPrevious < AccThresh2). AccThresh2 must be programmed to be greater than AccThresh1.
23:18	**AccThresh1: accuracy threshold level 1**. BIOS: 0. Throttle prefetches if (DramBwLevel == 1) and (AccCntPrevious < AccThresh1). AccThresh1 must be programmed to be greater than AccThresh0.
17:12	**AccThresh0: accuracy threshold level 0**. BIOS: 0. Throttle prefetches if (DramBwLevel == 0) and (AccCntPrevious < AccThresh0).
11:6	**TrackThreshold: prefetch throttling tracking threshold**. BIOS: 0Fh. Specifies how many tracking structure writes will occur before the current accuracy count (AccCntCurrent[5:0]) replaces the previous accuracy count (AccCntPrevious[5:0]).
5:0	**CaptureThreshold: prefetch throttling capture threshold**. BIOS: 0Fh. Specifies that 1 out of every CaptureThreshold prefetches that are sent to the NB will be inserted into the tracking structure.

MSRC001_1030 IBS Fetch Control (IbsFetchCtl)

Reset: 0000_0000_0000_0000h. See 2.6.2 [Instruction Based Sampling (IBS)].
The IBS fetch sampling engine is described as follows:

- The periodic fetch counter is an internal 20-bit counter:
 - The periodic fetch counter [19:4] is set to IbsFetchCnt[19:4] and the periodic fetch counter [3:0] is set according to IbsRandEn when IbsFetchEn is changed from 0 to 1.
 - It increments for every fetch cycle that completes when IbsFetchEn == 1 and IbsFetchVal == 0.
 - The periodic fetch counter is undefined when IbsFetchEn == 0 or IbsFetchVal == 1.
 - When IbsFetchCnt[19:4] is read it returns the current value of the periodic fetch counter [19:4].
- When the periodic fetch counter reaches {IbsFetchMaxCnt[19:4], 0h} and the selected instruction fetch completes or is aborted:
 - IbsFetchVal is set to 1.
 - Drivers can't assume that IbsFetchCnt[19:4] is 0 when IbsFetchVal == 1.
 - The status of the operation is written to the IBS fetch registers (this register, MSRC001_1031 and MSRC001_1032).
 - An interrupt is generated as specified by MSRC001_103A. The interrupt service routine associated with this interrupt is responsible for saving the performance information stored in IBS execution registers.

Bits	Description
63:59	Reserved.
58	**IbsFetchL2Miss: L2 cache miss for the sampled fetch**. Read-only; Cleared-by-hardware. 1=The instruction fetch missed in the L2 Cache. Qualified by (IbsFetchComp == 1).

57	**IbsRandEn: random instruction fetch tagging enable**. Read-write. 1=Bits[3:0] of the fetch counter are randomized when IbsFetchEn is set to start the fetch counter. 0=Bits[3:0] of the fetch counter are set to 0 when IbsFetchEn is set to start the fetch counter.
56	**IbsL2TlbMiss: instruction cache L2TLB miss**. Read-only; Set-by-hardware. 1=The instruction fetch missed in the L2 TLB.
55	**IbsL1TlbMiss: instruction cache L1TLB miss**. Read-only; Set-by-hardware. 1=The instruction fetch missed in the L1 TLB.
54:53	**IbsL1TlbPgSz: instruction cache L1TLB page size**. Read-only; Updated-by-hardware. Indicates the page size of the translation in the L1 TLB. This field is only valid if IbsPhyAddrValid == 1. Bits — Description 00b — 4 KB 01b — 2 MB 10b — 1 GB 11b — Reserved
52	**IbsPhyAddrValid: instruction fetch physical address valid**. Read-only; Set-by-hardware. 1=The physical address in MSRC001_1032 and the IbsL1TlbPgSz field are valid for the instruction fetch.
51	**IbsIcMiss: instruction cache miss**. Read-only; Set-by-hardware. 1=The instruction fetch missed in the instruction cache.
50	**IbsFetchComp: instruction fetch complete**. Read-only; Set-by-hardware. 1=The instruction fetch completed and the data is available for use by the instruction decoder.
49	**IbsFetchVal: instruction fetch valid**. Read-only; Set-by-hardware. 1=New instruction fetch data available. When this bit is set, the fetch counter stops counting and an interrupt is generated as specified by MSRC001_103A. This bit must be cleared for the fetch counter to start counting. When clearing this bit, software can write 0000h to IbsFetchCnt[19:4] to start the fetch counter at IbsFetchMaxCnt[19:4].
48	**IbsFetchEn: instruction fetch enable**. Read-write. 1=Instruction fetch sampling is enabled.
47:32	**IbsFetchLat: instruction fetch latency**. Read-only; Set-by-hardware. Indicates the number of clock cycles from when the instruction fetch was initiated to when the data was delivered to the core. If the instruction fetch is abandoned before the fetch completes, this field returns the number of clock cycles from when the instruction fetch was initiated to when the fetch was abandoned.
31:16	**IbsFetchCnt[19:4]**. Read-write; Updated-by-hardware. Provides read/write access to bits[19:4] of the periodic fetch counter. Programming this field to a value greater than or equal to IbsFetchMaxCnt[19:4] results in undefined behavior.
15:0	**IbsFetchMaxCnt[19:4]**. Read-write. Specifies bits[19:4] of the maximum count value of the periodic fetch counter. Programming this field to 0000h and setting IbsFetchEn results in undefined behavior. Bits[3:0] of the maximum count are always 0000b.

MSRC001_1031 IBS Fetch Linear Address (IbsFetchLinAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**IbsFetchLinAd: instruction fetch linear address**. Read-write; Updated-by-hardware. Provides the linear address in canonical form for the tagged instruction fetch.

MSRC001_1032 IBS Fetch Physical Address (IbsFetchPhysAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**IbsFetchPhysAd: instruction fetch physical address**. Read-write; Updated-by-hardware. Provides the physical address for the tagged instruction fetch. The lower 12 bits are not modified by address translation, so they are always the same as the linear address. This field contains valid data only if MSRC001_1030[IbsPhyAddrValid] is asserted.

MSRC001_1033 IBS Execution Control (IbsOpCtl)

Reset: 0000_0000_0000_0000h. See 2.6.2 [Instruction Based Sampling (IBS)].
The IBS execution sampling engine is described as follows for IbsOpCntCtl == 1. If IbsOpCntCtl == 1 then references to "periodic op counter" mean "periodic cycle counter".

- The periodic op counter is an internal 27-bit counter:
 - It is set to IbsOpCurCnt[26:0] when IbsOpEn is changed from 0 to 1.
 - It increments every dispatched op when IbsOpEn == 1 and IbsOpVal == 0.
 - The periodic op counter is undefined when IbsOpEn == 0 or IbsOpVal == 1.
 - When IbsOpCurCnt[26:0] is read then it returns the current value of the periodic micro-op counter [26:0].
- When the periodic micro-op counter reaches IbsOpMaxCnt:
 - The next dispatched micro-op is tagged if IbsOpCntCtl == 1. A valid op in the next dispatched line is tagged if IbsOpCntCtl == 0. See IbsOpCntCtl.
 - The periodic micro-op counter [26:7] = 0; [6:0] is randomized by hardware.
- The periodic micro-op counter is not modified when a tagged micro-op is flushed.
- When a tagged micro-op is retired:
 - IbsOpVal is set to 1.
 - Drivers can't assume that IbsOpCurCnt is 0 when IbsOpVal == 1.
 - The status of the operation is written to the IBS execution registers (this register, MSRC001_1034, MSRC001_1035, MSRC001_1036, MSRC001_1037, MSRC001_1038 and MSRC001_1039).
 - An interrupt is generated as specified by MSRC001_103A. The interrupt service routine associated with this interrupt is responsible for saving the performance information stored in IBS execution registers.

Bits	Description
63:59	Reserved.
58:32	**IbsOpCurCnt[26:0]: periodic op counter current count**. Read-write; Updated-by-hardware. Returns the current value of the periodic op counter.
31:27	Reserved.
26:20	**IbsOpMaxCnt[26:20]: periodic op counter maximum count**. Read-write. See IbsOpMaxCnt[19:4].
19	**IbsOpCntCtl: periodic op counter count control**. Read-write. 1=Count dispatched Micro-ops; when a roll-over occurs, the counter is preloaded with a pseudorandom 7-bit value between 1 and 127. 0=Count clock cycles; a 1-of-4 round-robin counter selects an op in the next dispatch line; if the op pointed to by the round-robin counter is invalid, then the next younger valid op is selected.
18	**IbsOpVal: micro-op sample valid**. Read-write; Set-by-hardware. 1=New instruction execution data available; the periodic op counter is disabled from counting. An interrupt may be generated when this bit is set as specified by MSRC001_103A[LvtOffset].

17	**IbsOpEn: micro-op sampling enable**. Read-write. 1=Instruction execution sampling enabled.
16	Reserved.
15:0	**IbsOpMaxCnt[19:4]: periodic op counter maximum count**. Read-write. IbsOpMaxCnt[26:0] = {IbsOpMaxCnt[26:20], IbsOpMaxCnt[19:4], 0000b}. Specifies maximum count value of the periodic op counter. Bits[3:0] of the maximum count are always 0000b. Bits Description 0008h-0000h Reserved FFFFh-0009h <IbsOpMaxCnt[19:4]*16> ops

MSRC001_1034 IBS Op Logical Address (IbsOpRip)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**IbsOpRip: micro-op linear address**. Read-write; Updated-by-hardware. Linear address in canonical form for the instruction that contains the tagged micro-op.

MSRC001_1035 IBS Op Data (IbsOpData)

Bits	Description
63:41	Reserved.
40	**IbsOpMicrocode**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation from microcode.
39	**IbsOpBrnFuse: fused branch micro-op**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation was a fused branch micro-op. Support indicated by CPUID Fn8000_001B_EAX[OpBrnFuse].
38	**IbsRipInvalid: RIP is invalid**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation RIP is invalid. Support indicated by CPUID Fn8000_001B_EAX[RipInvalidChk].
37	**IbsOpBrnRet: branch micro-op retired**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation was a branch micro-op that retired.
36	**IbsOpBrnMisp: mispredicted branch micro-op**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation was a branch micro-op that was mispredicted. Qualified by IbsOpBrnRet==1.
35	**IbsOpBrnTaken: taken branch micro-op**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation was a branch micro-op that was taken. Qualified by IbsOpBrnRet == 1.
34	**IbsOpReturn: return micro-op**. Read-write; Updated-by-hardware. Reset: 0. 1=Tagged operation was return micro-op. Qualified by IbsOpBrnRet == 1.
33:32	Reserved.
31:16	**IbsTagToRetCtr: micro-op tag to retire count**. Read-write; Updated-by-hardware. Reset: 0. This field returns the number of cycles from when the micro-op was tagged to when the micro-op was retired. This field is equal to IbsCompToRetCtr when the tagged micro-op is a NOP.
15:0	**IbsCompToRetCtr: micro-op completion to retire count**. Read-write; Updated-by-hardware. Reset: 0. This field returns the number of cycles from when the micro-op was completed to when the micro-op was retired.

MSRC001_1036 IBS Op Data 2 (IbsOpData2)

Reset: 0000_0000h. Northbridge data is only valid for load operations that miss both the L1 data cache and the L2 cache. If a load operation crosses a cache line boundary, the data returned in this register is the data for the access to the lower cache line.

Bits	Description
63:6	Reserved.
5	**NbIbsReqCacheHitSt**: **IBS cache hit state**. Read-write; Updated-by-hardware. Valid when the data source type is Cache(2h). 0=M State. 1=O State.
4	**NbIbsReqDstNode**: **IBS request destination node**. Read-write; Updated-by-hardware. 0=The request is serviced by the NB in the same node as the core. 1=The request is serviced by the NB in a different node than the core. Valid when NbIbsReqSrc is non-zero.
3	Reserved.
2:0	**NbIbsReqSrc: northbridge IBS request data source**. Read-write. Bits — Description 0h — No valid status 1h — Reserved 2h — Cache: data returned from another compute-unit cache. 3h — DRAM: data returned from DRAM 4h — Reserved for remote cache 5h — Reserved 6h — Reserved 7h — Other: data returned from MMIO/Config/PCI/APIC

MSRC001_1037 IBS Op Data 3 (IbsOpData3)

Reset: 0000_0000_0000_0000h. If a load or store operation crosses a 128-bit boundary, the data returned in this register is the data for the access to the data below the 128-bit boundary.

Bits	Description
63:48	**IbsTlbRefillLat: L1 DTLB refill latency**. Read-write; Updated-by-hardware. The number of cycles from when a L1 DTLB refill is triggered by a tagged op to when the L1 DTLB fill has been completed.
47:32	**IbsDcMissLat: data cache miss latency**. Read-write; Updated-by-hardware. Indicates the number of clock cycles from when a miss is detected in the data cache to when the data was delivered to the core. The value returned by this counter is not valid for data cache writes or prefetch instructions.
31:26	**IbsOpDcMissOpenMemReqs: outstanding memory requests on DC fill**. Read-write; Updated-by-hardware. The number of allocated, valid DC MABs when the MAB corresponding to a tagged DC miss op is deallocated. Includes the MAB allocated by the sampled op. 0_0000b=No information provided.

25:22	**IbsOpMemWidth: load/store size in bytes**. Read-write; Updated-by-hardware. Report the number of bytes the load or store is attempting to access. Bits — Description 0h — No information provided. 1h — Byte 2h — Word 3h — DW 4h — QW 5h — OW Fh-6h — Reserved
21	**IbsSwPf: software prefetch**. Read-write; Updated-by-hardware. 1=The operation is a software prefetch.
20	**IbsL2Miss: L2 cache miss for the sampled operation**. Read-write; Updated-by-hardware. 1=The operation missed in the L2, regardless of whether the operation initiated the request to the L2.
19	**IbsDcL2TlbHit1G: data cache L2TLB hit in 1G page**. Read-write; Updated-by-hardware. 1=The physical address for the tagged load or store operation was present in a 1G page table entry in the data cache L2TLB.
18	**IbsDcPhyAddrValid: data cache physical address valid**. Read-write; Updated-by-hardware. 1=The physical address in MSRC001_1039 is valid for the load or store operation.
17	**IbsDcLinAddrValid: data cache linear address valid**. Read-write; Updated-by-hardware. 1=The linear address in MSRC001_1038 is valid for the load or store operation.
16	**DcMissNoMabAlloc: DC miss with no MAB allocated**. Read-write; Updated-by-hardware. 1=The tagged load or store operation hit on an already allocated MAB.
15	**IbsDcLockedOp: locked operation**. Read-write; Updated-by-hardware. 1=Tagged load or store operation is a locked operation.
14	**IbsDcUcMemAcc: UC memory access**. Read-write; Updated-by-hardware. 1=Tagged load or store operation accessed uncacheable memory.
13	**IbsDcWcMemAcc: WC memory access**. Read-write; Updated-by-hardware. 1=Tagged load or store operation accessed write combining memory.
12:9	Reserved.
8	**IbsDcMisAcc: misaligned access**. Read-write; Updated-by-hardware. 1=The tagged load or store operation crosses a 128-bit address boundary.
7	**IbsDcMiss: data cache miss**. Read-write; Updated-by-hardware. 1=The cache line used by the tagged load or store was not present in the data cache.
6	**IbsDcL2tlbHit2M: data cache L2TLB hit in 2M page**. Read-write; Updated-by-hardware. 1=The physical address for the tagged load or store operation was present in a 2M page table entry in the data cache L2TLB.
5	**IbsDcL1TlbHit1G: data cache L1TLB hit in 1G page**. Read-write; Updated-by-hardware. 1=The physical address for the tagged load or store operation was present in a 1G page table entry in the data cache L1TLB.
4	**IbsDcL1TlbHit2M: data cache L1TLB hit in 2M page**. Read-write; Updated-by-hardware. 1=The physical address for the tagged load or store operation was present in a 2M page table entry in the data cache L1TLB.

3	**IbsDcL2TlbMiss: data cache L2TLB miss**. Read-write; Updated-by-hardware. 1=The physical address for the tagged load or store operation was not present in the data cache L2TLB.
2	**IbsDcL1tlbMiss: data cache L1TLB miss**. Read-write; Updated-by-hardware. 1=The physical address for the tagged load or store operation was not present in the data cache L1TLB.
1	**IbsStOp: store op**. Read-write; Updated-by-hardware. 1=Tagged operation is a store operation.
0	**IbsLdOp: load op**. Read-write; Updated-by-hardware. 1=Tagged operation is a load operation.

MSRC001_1038 IBS DC Linear Address (IbsDcLinAd)

Reset: 0000_0000_0000_0000h.

Bits	Description
63:0	**IbsDcLinAd**. Read-write; Updated-by-hardware. Provides the linear address in canonical form for the tagged load or store operation. This field contains valid data only if MSRC001_1037[IbsDcLinAddrValid] is asserted.

MSRC001_1039 IBS DC Physical Address (IbsDcPhysAd)

Bits	Description
63:48	RAZ.
47:0	**IbsDcPhysAd: load or store physical address**. Read-write; Updated-by-hardware. Reset: 0. Provides the physical address for the tagged load or store operation. The lower 12 bits are not modified by address translation, so they are always the same as the linear address. This field contains valid data only if MSRC001_1037[IbsDcPhyAddrValid] is asserted.

MSRC001_103A IBS Control

GP-write.

Bits	Description
63:9	Reserved.
8	**LvtOffsetVal: local vector table offset valid**. MSRC001_103A[LvtOffsetVal] is an alias of D18F3x1CC[LvtOffsetVal].
7:4	Reserved.
3:0	**LvtOffset: local vector table offset**. MSRC001_103A[LvtOffset] is an alias of D18F3x1CC[LvtOffset].

MSRC001_103B IBS Branch Target Address (BP_IBSTGT_RIP)

Reset: 0000_0000_0000_0000h. Support for this register indicated by CPUID Fn8000_001B_EAX[BrnTrgt].

Bits	Description
63:0	**IbsBrTarget**. Read-write; Updated-by-hardware. The logical address in canonical form for the branch target. Contains a valid target if non-0. Qualified by MSRC001_1035[IbsOpBrnRet] == 1.

MSRC001_103C IBS Fetch Control Extended (IC_IBS_EXTD_CTL)

Reset: 0000_0000_0000_0000h. Support for this register indicated by CPUID Fn8000_001B_EAX[IbsFetchCtlExtd].

Bits	Description
63:16	Reserved.
15:0	**IbsItlbRefillLat: ITLB Refill Latency for the sampled fetch, if there is a reload**. Read-only; Updated-by-hardware. The number of cycles when the fetch engine is stalled for an ITLB reload for the sampled fetch. If there is no reload, the latency is 0.

MSRC001_103D IBS Op Data 4 (DC_IBS_DATA2)

Reset: 0000_0000_0000_0000h. Support for this register indicated by CPUID Fn8000_001B_EAX[IbsOpData4].

Bits	Description
63:1	Reserved.
0	**IbsOpLdResync**. Read-write.

MSRC001_1090 Processor Feedback Constants 0

Read-write; Per-compute-unit.

Bits	Description
63:16	Reserved.
15:8	**RefCountScale**. Reset: Product-specific.
7:0	**ActualCountScale**. Reset: Product-specific.

MSRC001_10A1 Contention Blocking Buffer Control (CU_CBBCFG)

Per-compute-unit; Read-write.

Bits	Description
63:16	Reserved.
15:13	**TimeoutValueForCBBCLock**. Read-write. Reset: 0. BIOS: 001b. Specifies the timeout value set for CBB entries following cache lock release.

Bits	Description	Bits	Description
000b	32 cycles	100b	160 cycles
001b	64 cycles	101b	192 cycles
010b	96 cycles	110b	224 cycles
011b	128 cycles	111b	256 cycles

<table>
<tr><td>12:10</td><td>TimeoutValueForCBBAlloc. Read-write. Reset: 0. BIOS: 001b. Specifies the timeout value set when CBB entries are allocated in the UT pipe.
<table>
<tr><th>Bits</th><th>Description</th><th>Bits</th><th>Description</th></tr>
<tr><td>000b</td><td>32 cycles</td><td>100b</td><td>160 cycles</td></tr>
<tr><td>001b</td><td>64 cycles</td><td>101b</td><td>192 cycles</td></tr>
<tr><td>010b</td><td>96 cycles</td><td>110b</td><td>224 cycles</td></tr>
<tr><td>011b</td><td>128 cycles</td><td>111b</td><td>256 cycles</td></tr>
</table></td></tr>
<tr><td>9</td><td>AllocCBBForShared. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to be allocated on O/S data returns.</td></tr>
<tr><td>8</td><td>AllocCBBForExclusive. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to be allocated on M/Muw/E data returns.</td></tr>
<tr><td>7</td><td>AllocCBBForCLOCK. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to be allocated by CLOCK requests.</td></tr>
<tr><td>6</td><td>AllocCBBForPFLock. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to be allocated by PFLOCK requests.</td></tr>
<tr><td>5</td><td>AllocCBBForWccFill. Read-write. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to be allocated by WCC FILL requests.</td></tr>
<tr><td>4</td><td>AllocCBBForLsFill. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to be allocated by LS FILL requests.</td></tr>
<tr><td>3</td><td>Reserved.</td></tr>
<tr><td>2</td><td>EnCBBCLockHandling. Read-write. Reset: 0. BIOS: 1. Enables the CBB features that treat cache locks differently than other request types.</td></tr>
<tr><td>1</td><td>EnCBBCoreBlocking. Read-write. Reset: 0. BIOS: 1. Allows CBB entries to block core requests (in addition to probes).</td></tr>
<tr><td>0</td><td>MasterEn. Read-write. Reset: 0. BIOS: 1. 1=Enable CBB feature. CBB must be disabled before changes can be made to CBB configuration values. CBB may be enable by the same write that changes CBB configuration values.</td></tr>
</table>

3.24 Core Performance Counter Events

This section provides the core performance counter events that may be selected through MSRC001_020[A,8,6,4,2,0][EventSelect,UnitMask]. See that register and MSRC001_020[B,9,7,5,3,1] [Performance Event Counter (PERF_CTR[5:0])].

For NB performance counter events see 2.6.1.2 [NB Performance Monitor Counters] and 3.25 [NB Performance Counter Events].

3.24.1 PMCx0[1F:00] Events (FP)

PMCx000 FPU Pipe Assignment

PERF_CTL[3]. The number of operations (uops) and dual-pipe uops dispatched to each of the 3 FPU execution pipelines. This event reflects how busy the FPU pipelines are and may be used for workload characterization. This includes all operations performed by x87, MMX™, and SSE instructions, including moves. Each increment represents a one-cycle dispatch event. This event is a speculative event. (See PMCx0CB). Since this event includes non-numeric operations it is not suitable for measuring MFLOPS. The number of events logged per cycle can vary from 0 to 6 and must use PERF_CTL[3].

UnitMask	Description
7	Reserved.
6	Total number dual-pipe uops assigned to Pipe 2.
5	Total number dual-pipe uops assigned to Pipe 1.
4	Total number dual-pipe uops assigned to Pipe 0.
3	Reserved.
2	Total number uops assigned to Pipe 2.
1	Total number uops assigned to Pipe 1.
0	Total number uops assigned to Pipe 0.

PMCx001 FP Scheduler Empty

PERF_CTL[5:3]. This is a speculative event. The number of cycles in which the FPU scheduler is empty. Note that some ops like FP loads bypass the scheduler; see the FP MAS for the full list of “no pipe” ops that bypass the scheduler. Invert this (MSRC001_020[A,8,6,4,2,0][Inv] == 1) to count cycles in which at least one FPU operation is present in the FPU.

PMCx003 Retired SSE/AVX Operations

PERF_CTL[3]. This is a retire-based event. The number of retired SSE/AVX FLOPS. The number of events logged per cycle can vary from 0 to 32.

UnitMask	Description
7	Double precision multiply-add FLOPS. Multiply-add counts as 2 FLOPS.
6	Double precision divide/square root FLOPS.
5	Double precision multiply FLOPS.
4	Double precision add/subtract FLOPS.

3	Single precision multiply-add FLOPS. Multiply-add counts as 2 FLOPS.
2	Single-precision divide/square root FLOPS.
1	Single-precision multiply FLOPS.
0	Single-precision add/subtract FLOPS.

PMCx004 Number of Move Elimination and Scalar Op Optimization

PERF_CTL[3]. This is a dispatch based speculative event, and is useful for measuring the effectiveness of the Move elimination and Scalar code optimization schemes. The number of events logged per cycle can vary from 0 to 8 and must use PERF_CTL[3].

UnitMask	Description
7:4	Reserved.
3	Number of Scalar Ops optimized.
2	Number of Ops that are candidates for optimization (have Z-bit either set or pass).
1	Number of SSE Move Ops eliminated.
0	Number of SSE Move Ops.

PMCx005 Retired Serializing Ops

PERF_CTL[5:3]. The number of serializing Ops retired.

UnitMask	Description
7:4	Reserved.
3	x87 control word mispredict traps due to mispredictions in RC or PC, or changes in mask bits.
2	x87 bottom-executing uops retired.
1	SSE control word mispredict traps due to mispredictions in RC, FTZ or DAZ, or changes in mask bits.
0	SSE bottom-executing uops retired.

3.24.2 PMCx0[3F:20] Events (LS)

PMCx020 Segment Register Loads

PERF_CTL[5:0]. The number of segment register loads performed.

UnitMask	Description
7	Reserved.
6	HS
5	GS
4	FS
3	DS
2	SS
1	CS

0	ES

PMCx021 Pipeline Restart Due to Self-Modifying Code

PERF_CTL[5:0]. The number of pipeline restarts that were caused by self-modifying code (a store that hits any instruction that's been fetched for execution beyond the instruction doing the store).

PMCx022 Pipeline Restart Due to Various Events

PERF_CTL[5:0]. The number of pipeline restarts caused by events of various types, as selected by the mask.

UnitMask	Description
7:5	Reserved.
4	Resync due to other reasons (TLB guard violation, Fastpath op needing to be non-speculative).
3	Resync due to BLAST.
2	Resync due to overlapping store/load within dispatch group.
1	Resync due to load completing before older matching store.
0	Resync due to probe hitting a completed load.

PMCx023 Load Queue/Store Queue Full

PERF_CTL[2:0]. The number of cycles that the load queue (LDQ) or store queue (STQ) is full. The load queue holds loads that missed the data cache and are waiting on a refill; the store queue holds stores waiting to retire. This condition stalls further data cache accesses, although such stalls may be overlapped by independent instruction execution.

UnitMask	Description
7:2	Reserved.
1	The number of cycles that the store buffer is full.
0	The number of cycles that the load buffer is full.

PMCx024 Locked Operations

PERF_CTL[5:0]. This event covers locked operations performed and their non-speculative execution time.

PMCx026 Retired CLFLUSH Instructions

PERF_CTL[5:0]. The number of retired CLFLUSH instructions. This is a non-speculative event.

PMCx027 Retired CPUID Instructions

PERF_CTL[5:0]. The number of CPUID instructions retired.

PMCx029 LS Dispatch

PERF_CTL[5:0]. Counts the number of operations dispatched to the LS unit.

UnitMask	Description
7:3	Reserved.
2	Load-op-Stores.
1	Stores.
0	Loads.

PMCx02A Canceled Store to Load Forward Operations

PERF_CTL[5:0]. Counts the number of canceled store to load forward operations.

UnitMask	Description
7:2	Reserved.
1	Physical tag mismatch.
0	Either “store is smaller than load” or “different starting byte but partial overlap”.

PMCx02B SMIs Received

PERF_CTL[5:0]. Counts the number of SMIs received.

PMCx030 Executed CLFLUSH Instructions

PERF_CTL[5:0]. The number of executed CLFLUSH instructions. This is a speculative event.

PMCx032 Misaligned Stores

PERF_CTL[5:0]. The number of misaligned stores.

PMCx034 FP +Load Buffer Stall

PERF_CTL[5:0]. The number of loads stalled due to buffer full.

PMCx035 STLF

PERF_CTL[5:0]. Number of STLF hits.

3.24.3 PMCx0[5F:40] Events (DC)

PMCx040 Data Cache Accesses

PERF_CTL[5:0]. The number of accesses to the data cache for load and store references. This may include certain microcode scratchpad accesses, although these are generally rare. This event is a speculative event. The number of events logged per cycle can vary from 0 to 2.

PMCx041 Data Cache Misses

PERF_CTL[5:0]. The number of data cache references which missed in the data cache. This event is a speculative event. Only the first miss for a given line is included; access attempts by other instructions while the refill is still pending are not included in this event. Each event reflects one 64 B cache line refill, and counts of this event are the same as, or very close to, the combined count for PMCx042. The number of events logged per cycle can vary from 0 to 2.

UnitMask	Description
7:2	Reserved.
1	First streaming store to a 64 B cache line.
0	First data cache miss or streaming store to a 64 B cache line.

PMCx042 Data Cache Refills from L2 or System

PERF_CTL[5:0]. The number of data cache refills satisfied from the L2 cache and/or the system. Each increment reflects a 64 B transfer. This event is a speculative event.

UnitMask	Description
7:5	Reserved.
4	No-acknowledge fill response.
3	Fill with read data error.
2	Reserved.
1	Early valid status turned out to be invalid.
0	Fill with good data. (Final valid status is valid)

PMCx043 Data Cache Refills from System

PERF_CTL[2:0]. The number of L1 cache refills satisfied from the system (system memory or another cache), as opposed to the L2. Each increment reflects a 64 B transfer. This event is a speculative event.

PMCx045 Unified TLB Hit

PERF_CTL[2:0]. The number of TLB accesses that miss in the L1 DTLB or L1 and L2 ITLBs and hit in the unified TLB (UCTLB). This event is a speculative event.

UnitMask	Description
7	Reserved.
6	1 GB unified TLB hit for instruction.
5	2 MB unified TLB hit for instruction.
4	4 KB unified TLB hit for instruction.
3	Reserved.
2	1 GB unified TLB hit for data.
1	2 MB unified TLB hit for data.
0	4 KB unified TLB hit for data.

PMCx046 Unified TLB Miss

PERF_CTL[2:0]. The number of TLB accesses that miss in all TLBs. This event is a speculative event.

UnitMask	Description
7	Reserved.
6	1 GB unified TLB miss for instruction.
5	2 MB unified TLB miss for instruction.
4	4 KB unified TLB miss for instruction.
3	Reserved.
2	1 GB unified TLB miss for data.
1	2 MB unified TLB miss for data.
0	4 KB unified TLB miss for data.

PMCx047 Misaligned Accesses

PERF_CTL[5:0]. The number of data cache accesses that are misaligned. These are accesses which cross an 8-B boundary. They incur an extra cache access (reflected in PMCx040), and an extra cycle of latency on reads. This event is a speculative event.

PMCx04B Prefetch Instructions Dispatched

PERF_CTL[5:0]. The number of prefetch instructions dispatched by the decoder. Such instructions may or may not cause a cache line transfer. Any Dcache and L2 accesses, hits and misses by prefetch instructions are included in these types of events. This event is a speculative event.

UnitMask	Description
7:3	Reserved.
2	NTA (PrefetchNTA).
1	Store (PrefetchW).
0	Load (Prefetch, PrefetchT0/T1/T2).

PMCx052 Ineffective Software Prefetchs

PERF_CTL[5:0]. The number of software prefetches that did not fetch data outside of the processor core.

UnitMask	Description
7:4	Reserved.
3	Software prefetch hit in the L2.
2:1	Reserved.
0	Software prefetch hit in the L1.

3.24.4 PMCx[1:0][7F:60] Events (CU)

PMCx060 Command Related to Victim Buffers

PERF_CTL[2:0].

UnitMask	Description
7	Lock.
6:5	Reserved.
4	Clean Victim Command.
3	Write Victim Block.
2:0	Reserved.

PMCx061 Command Related to Masked Operations

PERF_CTL[2:0]. Count Masked Byte and DW reads and writes to the NB. Byte sized read and write commands can request the transfer of up to 32 bytes. DW sized read and write commands can request the transfer of up to 32 DW's. Combining of the WC memory type can cause 1 B/DW write to represent multiple stores.

UnitMask	Description
7:6	Reserved.
5	Write Double-word.
4	Write Byte.
3	Reserved.
2	Read Double-word.
1	Reserved.
0	Read Byte.

PMCx062 Command Related to Read Block Operations

PERF_CTL[2:0].

UnitMask	Description
7	Reserved.
6	Read Block Speculative Shared.
5	RdBlkSpecMod.
4	RdBlkSpec.
3	Reserved.
2	Read Block Shared.
1	RdBlkMod.
0	Read Block.

PMCx063 Command Related to Change to Dirty Operations

PERF_CTL[2:0].

UnitMask	Description
7:5	Reserved.
4	Change to Dirty.
3:0	Reserved.

PMCx064 Dram System Request

PERF_CTL[2:0].

PMCx065 Memory Requests by Type

PERF_CTL[2:0]. These events reflect accesses to uncacheable (UC), write-combining (WC), and streaming store (SS) activity to WB memory.

UnitMask	Description
7	Requests to non-cacheable (WC+/SS, but not WC) memory, consisting of reads and 64-B sized buffer flushes.
6:2	Reserved.
1	Requests to non-cacheable (WC, but not WC+/SS) memory, consisting of reads and 64 B sized buffer flushes.
0	Requests to non-cacheable (UC) memory.

PMCx067 Data Cache Prefetches

UnitMask	Description
7:2	Reserved.
1	Prefetch attempts.
0	Prefetch Promotion.

PMCx068 MAB Requests

PERF_CTL[2:0]. Events PMCx068 and PMCx069 reflect utilization of the Miss Address buffers (MABs), which handle IC, DC, TLB, WCC, and WCB related requests. The UnitMask[BufferID] is an encoded value which selects one of the MABs. PMCx068 counts the number of cacheable L2 misses handled by the selected MAB; PMCx069 counts the number of cycles the selected MAB is busy waiting for the NB response. The average latency seen by the selected MAB is the number of cycles spent waiting (PMCx069) divided by the number of requests (PMCx068).

UnitMask	Description
7:0	**BufferID**. Bits: Description 27-0: MAB ID. 255-28: Reserved.

PMCx069 MAB Wait Cycles

PERF_CTL[2:0].
See PMCx068.

UnitMask	Description
7:0	**BufferID**. See: PMCx068[BufferID].

PMCx06C System Response by Coherence State

PERF_CTL[2:0]. The number of responses from the system for cache refill requests. The UnitMask may be used to select specific cache coherency states. Each increment represents one 64 B cache line transferred from the system (DRAM or another cache, including another core on the same node) to the data cache, instruction cache or L2 cache (for data prefetcher and TLB table walks). Modified-state responses may be for Dcache store miss refills, PrefetchW software prefetches, hardware prefetches for a store-miss stream, or Change-to-Dirty requests that get a dirty (Owned) probe hit in another cache. Exclusive responses may be for any Icache refill, Dcache load miss refill, other software prefetches, hardware prefetches for a load-miss stream, or TLB table walks that miss in the L2 cache; Shared responses may be for any of those that hit a clean line in another cache.

UnitMask	Description
7:6	Reserved.
5	Modified unwritten.
4	Data Error.
3	Owned.
2	Shared.
1	Modified.
0	Exclusive.

PMCx06D Octwords Written to System

PERF_CTL[2:0]. The number of OW (16 B) data transfers from the processor to the system. These may be part of a 64-B cache line writeback or a 64-B dirty probe hit response, each of which would cause four increments; or a partial or complete Write Combining buffer flush (Sized Write), which could cause from one to four increments.

UnitMask	Description
7:6	Reserved.

5:1	Reserved.
0	OW write transfer.

PMCx075 Cache Cross-invalidates

These reflect internal probes for Icache or Dcache misses that hit in the Dcache or Icache, causing the line to be invalidated. These may result from code modification, data being located too close to code, or virtual address aliasing. The aliasing cases arise when a physical memory location is referenced via two or more virtual addresses which differ in bits[14:12]. Such aliasing cases are generally uncommon.

UnitMask	Description
7:4	Reserved.
3	IC Invalidates DC (execution of recently modified code, or modified data too close to code).
2	IC Invalidates IC (aliasing).
1	DC Invalidates DC (aliasing).
0	DC Invalidates IC (modification of cached instructions, or of data located too close to code).

PMCx076 CPU Clocks not Halted

PERF_CTL[5:0].
The number of clocks that the CPU is not in a halted state (due to STPCLK or a HLT instruction). Note: this event allows system idle time to be automatically factored out from IPC (or CPI) measurements, providing the OS halts the CPU when going idle. If the OS goes into an idle loop rather than halting, such calculations are influenced by the IPC of the idle loop.

PMCx07D Requests to L2 Cache

PERF_CTL[2:0]. The number of requests to the L2 cache for Icache or Dcache fills, or page table lookups for the TLB. These events reflect only read requests to the L2; writes to the L2 are indicated by PMCx07E. See PMCx081, PMCx082, PMCx083, PMCx041, PMCx042, PMCx043.

UnitMask	Description
7	Reserved.
6	L2 cache prefetcher request.
5	Reserved.
4	Canceled request.
3	NB probe request.
2	TLB fill (page table walks).
1	DC fill.
0	IC fill.

PMCx07E L2 Cache Misses

PERF_CTL[2:0]. The number of requests that miss in the L2 cache. This may include some amount of speculative activity. The IC-fill-miss and DC-fill-miss events tend to mirror the Icache and Dcache refill-from-sys-

tem PMCx083 and PMCx043, and tend to include more speculative activity than those events.

UnitMask	Description
7:6	Reserved.
5	Reserved.
4	L2 Cache Prefetcher request.
3	Reserved.
2	TLB page table walk.
1	DC fill (includes possible replays, whereas PMCx041 does not).
0	IC fill.

PMCx07F L2 Fill/Writeback

PERF_CTL[2:0]. Each increment represents a 64-B cache line transfer.

UnitMask	Description
7:3	Reserved.
2	**L2CleanWritebacks**. L2 Clean Writebacks to system.
1	**L2Writebacks**. L2 Writebacks to system (Clean and Dirty).
0	**L2Fills**. L2 fills from system. Note: Fills for non-temporal software prefetch and WP-memtype fills also are counted in this event even though they don't get cached in L2.

PMCx165 Page Splintering

PERF_CTL[2:0]. Counts the number of TLB reloads where a large page is installed into the TLB as a smaller page size.

UnitMask	Description
7:3	Reserved.
2	Host page size is larger than the guest page size.
1	Splintering due to MTRRs, IORRs, APIC, TOMs or other special address region.
0	Guest page size is larger than the host page size when nested paging is enabled.

PMCx16C L2 Prefetcher Trigger Events

PERF_CTL[2:0].

UnitMask	Description
7:2	Reserved.
1	Store L1 miss seen by prefetcher.
0	Load L1 miss seen by prefetcher.

PMCx177 XAB Allocation Stall

PERF_CTL[2:0]. Counts cycles that XAB allocation is stalled due to full condition. The event counts regardless of whether there are any ops that need to be allocated to the XAB.

UnitMask	Description
7:2	Reserved
1	XAB Allocation Throttled - Count number of cycles that the XAB is almost full and allocation to XAB is slowed down to one allocation per 3 cycles.
0	XAB Allocation Stalled - Count number of cycles that the XAB is full and cannot accept another operation.

PMCx17F Number of free XAB entries available to thread

PERF_CTL[0]. This is a 5-bit multi-event that increments each cycle by the number of free entries in the XAB.

UnitMask	Description
7:1	Reserved.
0	Count free entries in the XAB.

3.24.5 PMCx[1:0][9F:80] Events (IC)

Note: All instruction cache events are speculative events unless specified otherwise.

PMCx080 Instruction Cache Fetches

PERF_CTL[2:0].
The number of successful instruction cache accesses by the instruction fetcher that result in data being sent to the decoder. Each access is an aligned 32 byte read, from which a varying number of instructions may be decoded.

PMCx081 Instruction Cache Misses

PERF_CTL[2:0].
The number of instruction fetches and prefetch requests that miss in the instruction cache. This is typically equal to or very close to the sum of events 82h and 83h. Each miss results in a 64-byte cache line refill.

PMCx082 Instruction Cache Refills from L2

PERF_CTL[2:0].
The number of instruction cache refills satisfied from the L2 cache. Each increment represents one 64-byte cache line transfer.

PMCx083 Instruction Cache Refills from System

PERF_CTL[2:0].
The number of instruction cache refills from system memory (or another cache). Each increment represents one 64-byte cache line transfer.

PMCx084 L1 ITLB Miss, L2 ITLB Hit

PERF_CTL[2:0].
The number of instruction fetches that miss in the L1 ITLB but hit in the L2 ITLB.

PMCx085 L1 ITLB Miss, L2 ITLB Miss

PERF_CTL[2:0]. The number of instruction fetches that miss in both the L1 and L2 TLBs.

UnitMask	Description
7:3	Reserved.
2	Instruction fetches to a 1 GB page.
1	Instruction fetches to a 2 MB page.
0	Instruction fetches to a 4 KB page.

PMCx086 Pipeline Restart Due to Instruction Stream Probe

PERF_CTL[2:0]. The number of pipeline restarts caused by invalidating probes that hit on the instruction stream currently being executed. This would happen if the active instruction stream was being modified by another processor in an MP system - typically a highly unlikely event.

PMCx087 Instruction Fetch Stall

PERF_CTL[2:0]. The number of cycles the instruction fetcher is stalled for the core. This may be for a variety of reasons such as branch predictor updates, unconditional branch bubbles, far jumps and cache misses, instruction fetching for the other core while instruction fetch for this core is stalled, among others. May be overlapped by instruction dispatch stalls or instruction execution, such that these stalls don't necessarily impact performance.

PMCx088 Return Stack Hits

PERF_CTL[2:0]. The number of near return instructions (RET or RET Iw) that get their return address from the return address stack (i.e., where the stack has not gone empty) for the core. This may include cases where the address is incorrect (return mispredicts). This may also include speculatively executed false-path returns. Return mispredicts are typically caused by the return address stack underflowing, however they may also be caused by an imbalance in calls vs. returns, such as doing a call but then popping the return address off the stack.

This event cannot be reliably compared with events C9h and CAh (such as to calculate percentage of return mispredicts due to an empty return address stack), since it may include speculatively executed false-path returns that are not included in those retire-time events.

PMCx089 Return Stack Overflows

PERF_CTL[2:0]. The number of (near) call instructions that cause the return address stack to overflow. When this happens, the oldest entry is discarded. This count may include speculatively executed calls.

PMCx08B Instruction Cache Victims

PERF_CTL[2:0]. The number of cachelines evicted from the instruction cache that cause an L2 write due to changed predecode (start bits); the L2 write due to changed predecode doesn't write the instruction bytes. This event does not count IC evictions with unchanged predecode, which are silently dropped without an L2 write. This event is not core specific and for either core counts the IC victims caused by both cores of the compute unit.

PMCx08C Instruction Cache Lines Invalidated

PERF_CTL[2:0]. The number of instruction cache lines invalidated. A non-SMC event is CMC (cross modifying code), either from the other core of the compute unit or another compute compute unit.

UnitMask	Description
7:4	Reserved.
3	SMC invalidating probe that hit on in-flight instructions.
2	SMC invalidating probe that missed on in-flight instructions.
1	Non-SMC invalidating probe that hit on in-flight instructions.
0	Non-SMC invalidating probe that missed on in-flight instructions.

PMCx099 ITLB Reloads

PERF_CTL[2:0]. The number of ITLB reload requests.

PMCx09A ITLB Reloads Aborted

PERF_CTL[2:0]. The number of ITLB reloads aborted.

PMCx186 Uops Dispatched From Decoder

PERF_CTL[2:0]. Counts uops that are dispatched from the decoder each cycle. The number of events logged per cycle can vary from 0 to 4. Note that when microcode dispatches between 1 to 4 uops in a cycle then 4 uops are counted in that cycle.

UnitMask	Description
7:1	Reserved.
0	**UopsDispatched: Uops Dispatched From Decoder**. **AllOps**. Ops dispatched from the decoder. Bits: Description 0h: AllOps: Ops dispatched from the decoder.

3.24.6 PMCx[1,0][DF:C0] Events (EX, DE)

PMCx0C0 Retired Instructions

PERF_CTL[5:0]. The number of instructions retired (execution completed and architectural state updated). This count includes exceptions and interrupts - each exception or interrupt is counted as one instruction.

PMCx0C1 Retired uops

PERF_CTL[5:0]. The number of micro-ops retired. This includes all processor activity (instructions, exceptions, interrupts, microcode assists, etc.). The number of events logged per cycle can vary from 0 to 4.

PMCx0C2 Retired Branch Instructions

PERF_CTL[5:0]. The number of branch instructions retired. This includes all types of architectural control flow changes, including exceptions and interrupts.

PMCx0C3 Retired Mispredicted Branch Instructions

PERF_CTL[5:0]. The number of branch instructions retired, of any type, that were not correctly predicted. This includes those for which prediction is not attempted (far control transfers, exceptions and interrupts).

PMCx0C4 Retired Taken Branch Instructions

PERF_CTL[5:0]. The number of taken branches that were retired. This includes all types of architectural control flow changes, including exceptions and interrupts.

PMCx0C5 Retired Taken Branch Instructions Mispredicted

PERF_CTL[5:0]. The number of retired taken branch instructions that were mispredicted.

PMCx0C6 Retired Far Control Transfers

PERF_CTL[5:0]. The number of far control transfers retired including far call/jump/return, IRET, SYSCALL and SYSRET, plus exceptions and interrupts. Far control transfers are not subject to branch prediction.

PMCx0C7 Retired Branch Resyncs

PERF_CTL[5:0]. The number of resync branches. These reflect pipeline restarts due to certain microcode assists and events such as writes to the active instruction stream, among other things. Each occurrence reflects a restart penalty similar to a branch mispredict. This is relatively rare.

PMCx0C8 Retired Near Returns

PERF_CTL[5:0]. The number of near return instructions (RET or RET Iw) retired.

PMCx0C9 Retired Near Returns Mispredicted

PERF_CTL[5:0]. The number of near returns retired that were not correctly predicted by the return address predictor. Each such mispredict incurs the same penalty as a mispredicted conditional branch instruction.

PMCx0CA Retired Mispredicted Taken Branch Instructions due to Target Mismatch

PERF_CTL[5:0]. The number of indirect branch instructions retired where the target address was not correctly predicted.

PMCx0CB Retired MMX™/FP Instructions

PERF_CTL[5:0]. The number of MMX™, SSE or x87 instructions retired. The UnitMask allows the selection of the individual classes of instructions as given in the table. Each increment represents one complete instruction. Since this event includes non-numeric instructions it is not suitable for measuring MFLOPS.

UnitMask	Description
7:3	Reserved.
2	SSE instructions (SSE, SSE2, SSE3, SSSE3, SSE4A, SSE4.1, SSE4.2, AVX, XOP, FMA4).
1	MMX™ instructions.
0	x87 instructions.

PMCx0CD Interrupts-Masked Cycles

PERF_CTL[5:0]. The number of cycles where interrupts are masked (EFLAGS.IF == 0). Using edge-counting with this event gives the number of times IF is cleared; dividing the cycle-count value by this value gives the average length of time that interrupts are disabled on each instance. Compare the edge count with PMCx0CF to determine how often interrupts are disabled for interrupt handling vs. other reasons (e.g., critical sections).

PMCx0CE Interrupts-Masked Cycles with Interrupt Pending

PERF_CTL[5:0]. The number of cycles where interrupts are masked (EFLAGS.IF == 0) and an interrupt is pending. Using edge-counting with this event and comparing the resulting count with the edge count for PMCx0CD gives the proportion of interrupts for which handling is delayed due to prior interrupts being serviced, critical sections, etc. The cycle count value gives the total amount of time for such delays. The cycle count divided by the edge count gives the average length of each such delay.

PMCx0CF Interrupts Taken

PERF_CTL[5:0]. The number of hardware interrupts taken. This does not include software interrupts (INT n instruction).

PMCx0D0 Decoder Empty

PERF_CTL[2:0]. The number of cycles where the decoder has nothing to dispatch (typically waiting on an instruction fetch that missed the Icache, or for the target fetch after a branch mispredict).

PMCx0D1 Dispatch Stalls

PERF_CTL[2:0]. The number of cycles where the decoder is stalled for any reason (has one or more instructions ready but can't dispatch them due to resource limitations in execution). This event counts even when dispatch selects the other core of the compute-unit. This is the combined effect of events PMCx0D3 to PMCx0D9, some of which may overlap; this event reflects the net stall cycles. The more common stall conditions (events PMCx0D5, PMCx0D6, PMCx0D7, PMCx0D8) may overlap considerably. The occurrence of these stalls is highly dependent on the nature of the code being executed (instruction mix, memory reference patterns, etc.).

PMCx0D3 Microsequencer Stall due to Serialization

PERF_CTL[2:0]. The number of cycles the microsequencer is stalled due to a serializing operation, which waits for the execution pipeline to drain. Relatively rare; mainly associated with system instructions. See PMCx0D1.

PMCx0D5 Dispatch Stall for Instruction Retire Queue Full

PERF_CTL[2:0]. The number of cycles the decoder is stalled because the instruction retire Q is full. This event counts even when dispatch selects the other core of the compute-unit. May occur simultaneously with certain other stall conditions; see PMCx0D1.

PMCx0D6 Dispatch Stall for Integer Scheduler Queue Full

PERF_CTL[2:0]. The number of cycles the decoder is stalled because a required integer unit scheduler queue is full. This event counts even when dispatch selects the other core of the compute-unit. May occur simultaneously with certain other stall conditions; see PMCx0D1.

PMCx0D7 Dispatch Stall for FP Scheduler Queue Full

PERF_CTL[2:0]. The number of cycles the decoder is stalled because the scheduler for the Floating Point scheduler queue is full. This event counts even when dispatch selects the other core of the compute-unit. This condition can be caused by a lack of parallelism in FP-intensive code, or by cache misses on FP operand loads (which could also show up as PMCx0D8 instead, depending on the nature of the instruction sequences). May occur simultaneously with certain other stall conditions; see PMCx0D1.

PMCx0D8 Dispatch Stall for LDQ Full

PERF_CTL[2:0]. The number of cycles the decoder is stalled because the load queue is full. This event counts even when dispatch selects the other core of the compute-unit. This generally occurs due to heavy cache miss activity. May occur simultaneously with certain other stall conditions; see PMCx0D1.

PMCx0D9 Microsequencer Stall Waiting for All Quiet

PERF_CTL[2:0]. The number of cycles the microsequencer is stalled waiting for all outstanding requests to the system to be resolved. Relatively rare; associated with certain system instructions and types of interrupts. May partially overlap certain other stall conditions; see PMCx0D1.

PMCx0DB FPU Exceptions

PERF_CTL[5:0]. The number of floating point unit exceptions for microcode assists. The UnitMask may be used to isolate specific types of exceptions.

UnitMask	Description
7:5	Reserved.
4	Bypass faults.
3	Ext2Int faults.
2	Int2Ext faults.

1	Total microtraps.
0	Total microfaults.

PMCx0D[F:C] DR[3:0] Breakpoint Matches

PERF_CTL[5:0].

Table 270: Register Mapping for PMCx0D[F:C]

Register	Function
PMCx0DC	DR0
PMCx0DD	DR1
PMCx0DE	DR2
PMCx0DF	DR3

The number of matches on the address in breakpoint register DR[3:0], per the breakpoint type specified in DR7. Matches occur if the access becomes becomes non-speculative, but not necessarily retired. Each instruction breakpoint match incurs an overhead of about 120 cycles; load/store breakpoint matches do not incur any overhead.

PMCx1C0 Retired x87 Floating Point Operations

PERF_CTL[5:3]. The number of x87 floating point ops that have retired.

UnitMask	Description
7:3	Reserved.
2	Divide and square root operations.
1	Multiply operations.
0	Add/subtract operations.

PMCx1CF Tagged IBS Ops

PERF_CTL[5:0].

UnitMask	Description
7:3	Reserved.
2	Number of times an operation could not be tagged by IBS because of a previous tagged operation that has not retired.
1	Number of operations tagged by IBS that retired.
0	Number of operations tagged by IBS.

PMCx1D0 Retired Fused Branch Instructions

PERF_CTL[5:0]. Implemented by EX. The number of fused retired branch instructions retired per cycle. The number of events logged per cycle can vary from 0 to 3.

PMCx1D8 Dispatch Stall for STQ Full

PERF_CTL[5:0]. The number of cycles the decoder is stalled because the store queue is full. This event counts even when dispatch selects the other core of the compute-unit. This generally occurs due to heavy cache miss activity. May occur simultaneously with certain other stall conditions.

PMCx1DD Cycles Without Dispatch Due To Integer PRF Tokens

PERF_CTL[2:0]. The number of cycles a core has valid operations for dispatch and one of the reasons for a dispatch stall is the absence of sufficient integer PRF tokens. This event counts even when dispatch selects the other core of the compute-unit.

PMCx1DE Cycles Without Dispatch Due to FP PRF Tokens

PERF_CTL[2:0]. The number of cycles a core has valid operations for dispatch and one of the reasons for a dispatch stall is the absence of sufficient FP PRF tokens. This event requires that the other core of the compute unit is in the Halt state. This event counts even when dispatch selects the other core of the compute-unit.

PMCx1DF FP Dispatch Contention

PERF_CTL[2:0]. Cycles in which there is contention between the two threads for FP dispatch.

UnitMask	Description
7:4	Reserved.
3	Counts operations stalled by lost FP dispatch arbitration while there was no FP dispatch on the other thread 1.
2	Counts operations stalled by lost FP dispatch arbitration while there was FP dispatch on the other thread 1.
1:0	**ContentionSel**. Bits — Description 00b — The other selected thread did not dispatch; this not selected thread could not have dispatched. 01b — The other selected thread did not dispatch; this not selected thread could have dispatched. 10b — The other selected thread did dispatch; this not selected thread could not have dispatched. 11b — The other selected thread did dispatch; this not selected thread could have dispatched.

3.25 NB Performance Counter Events

This section provides the performance counter events that may be selected through MSRC001_024[6,4,2,0][EventSelect,UnitMask]. See that register and MSRC001_024[7,5,3,1] [Northbridge Performance Event Counter (NB_PERF_CTR[3:0])].

Performance monitor events NBPMCx3EC, NBPMCx3ED, NBPMCx3EE, NBPMCx3EF are enabled by also programming D18F2x1[E8,E0,D8,D0]_dct[0], D18F2x1[EC,E4,DC,D4]_dct[0] with the EventSelect and UnitMask. Program EventSelect and UnitMask to zero to disable the performance monitor event. The resulting counter value is the OR function of all enabled DCTs on a particular NB performance monitor.

3.25.1 PMCx0E[7:4] Events (Memory Controller)

NBPMCx0E4 Memory Controller Bypass Counter Saturation

UnitMask	Description
7:6	Reserved.
5	DCQ Bypass Saturated. The DCT is selected by the field NBPMCx0E4[4:2].
4:2	Select DCQ bypass: Bits — Description 000b — Select DCT0 DCQ bypass 001b — Select DCT1 DCQ bypass 010b — Select DCT2 DCQ bypass 011b — Select DCT3 DCQ bypass 100b — Select DCT4 DCQ bypass 101b — Select DCT5 DCQ bypass 110b — Select DCT6 DCQ bypass 111b — Select DCT7 DCQ bypass
1	Memory controller medium priority bypass.
0	Memory controller high priority bypass.

3.25.2 PMCx0E[F:8] Events (Crossbar)

NBPMCx0E8 Thermal Status

UnitMask	Description
7	Reserved.
6	Number of clocks HTC P-state is active.
5	Number of clocks HTC P-state is inactive.
4:3	Reserved.
2	Number of times the HTC trip point is crossed.
1:0	Reserved.

NBPMCx0E9 CPU or IO Requests to Memory or IO

These events reflect request flow between units and nodes, as selected by the UnitMask. The UnitMask is divided into two fields: request type (CPU or IO access to IO or Memory) and source/target location (local vs. remote). One or more requests types must be enabled via bits[3:0], and at least one source and one target location must be selected via bits[7:4]. Each event reflects a request of the selected type(s) going from the selected source(s) to the selected target(s).

Not all possible paths are supported. The following table shows the UnitMask values that are valid for each request type:

Source/Target	CPU to Mem	CPU to IO	IO to Mem	IO to IO
Local -> Local	A8h	A4h	A2h	A1h

Source/Target	CPU to Mem	CPU to IO	IO to Mem	IO to IO
Local -> Remote	98h	94h	92h	91h
Remote -> Local	-	64h	-	61h
Remote -> Remote	-	-	-	-

Any of the mask values shown may be logically ORed to combine the events. For instance, local CPU requests to both local and remote nodes would be A8h | 98h = B8h. Any CPU to any IO would be A4h | 94h | 64h = F4h (but remote CPU to remote IO requests would not be included).

Note: It is not possible to tell from these events how much data is going in which direction, as there is no distinction between reads and writes. Also, particularly for IO, the requests may be for varying amounts of data, anywhere from one to sixty-four bytes. For a direct measure of the amount and direction of data flowing between nodes, use events F6h, F7h and F8h.

UnitMask	Description
7	From local node.
6	From remote node.
5	To local node.
4	To remote node.
3	CPU to Mem.
2	CPU to IO.
1	IO to Mem.
0	IO to IO.

NBPMCx0EA Cache Block Commands

The number of requests made to the system for cache line transfers or coherency state changes, by request type. Each increment represents one cache line transfer, except for Change-to-Dirty. If a Change-to-Dirty request hits on a line in another processor's cache that's in the Owned state, it causes a cache line transfer, otherwise there is no data transfer associated with Change-to-Dirty requests.

UnitMask	Description
7:6	Reserved.
5	Change-to-Dirty (first store to clean block already in cache).
4	Read Block Modified (Dcache store miss refill).
3	Read Block Shared (Icache refill).
2	Read Block (Dcache load miss refill).
1	Reserved.
0	Victim Block (Writeback).

NBPMCx0EB Sized Commands

The number of Sized Read/Write commands handled by the System Request Interface (local processor and hostbridge interface to the system). These commands may originate from the processor or hostbridge. Typical

uses of the various Sized Read/Write commands are given in the UnitMask table. See NBPMCx0EC, which provides a separate measure of Hostbridge accesses.

UnitMask	Description
7:6	Reserved.
5	SzRd DW (1-16 DWORDs). Typical Usage: Block-oriented DMA reads, typically cache-line size.
4	SzRd Byte (4 bytes). Typical Usage: Legacy or mapped IO.
3	Posted SzWr DW (1-16 DWORDs). Typical Usage: Block-oriented DMA writes, often cache-line sized; also processor Write Combining buffer flushes.
2	Posted SzWr Byte (1-32 bytes). Typical Usage: Sub-cache-line DMA writes, size varies; also flushes of partially-filled Write Combining buffer.
1	Non-Posted SzWr DW (1-16 DWORDs). Typical Usage: Legacy or mapped IO, typically 1 DWORD.
0	Non-Posted SzWr Byte (1-32 bytes). Typical Usage: Legacy or mapped IO, typically 1-4 bytes.

NBPMCx0EC Probe Responses and Upstream Requests

This covers two unrelated sets of events: cache probe results, and requests received by the hostbridge from devices on non-coherent links.

Probe results: These events reflect the results of probes sent from a memory controller to local caches. They provide an indication of the degree data and code is shared between processors (or moved between processors due to process migration). The dirty-hit events indicate the transfer of a 64-byte cache line to the requestor (for a read or cache refill) or the target memory (for a write). The system bandwidth used by these, in terms of bytes per unit of time, may be calculated as 64 times the event count, divided by the elapsed time. Sized writes to memory that cover a full cache line do not incur this cache line transfer -- they simply invalidate the line and are reported as clean hits. Cache line transfers occur for Change2Dirty requests that hit cache lines in the Owned state. (Such cache lines are counted as Modified-state refills for PMCx06C, System Read Responses.)

Upstream requests: The upstream read and write events reflect requests originating from a device on a local IO link.

UnitMask	Description
7	Upstream non-ISOC writes.
6	Upstream ISOC writes.
5	Upstream non-display refresh reads.
4	Upstream display refresh/ISOC reads.
3	Probe hit dirty with memory cancel (probed by DMA read or cache refill request).
2	Probe hit dirty without memory cancel (probed by Sized Write or Change2Dirty).
1	Probe hit clean.
0	Probe miss.

3.25.3 PMCx0F[F:0] Events (Crossbar)

3.25.4 NBPMCx1E[F:0] Events (Crossbar)

NBPMCx1E0 CPU to DRAM Requests to Target Node

This event counts all DRAM reads and writes generated by cores on the local node to the targeted node in the coherent fabric. This counter can be used to observe processor data affinity in NUMA aware operating systems.

UnitMask	Description
7	From Local node to Node 7.
6	From Local node to Node 6.
5	From Local node to Node 5.
4	From Local node to Node 4.
3	From Local node to Node 3.
2	From Local node to Node 2.
1	From Local node to Node 1.
0	From Local node to Node 0.

NBPMCx1E1 IO to DRAM Requests to Target Node

This event counts all DRAM reads and writes generated by IO devices attached to the IO links of the local node the targeted node in the coherent fabric. This counter can be used to observe IO device data affinity in NUMA aware operating systems.

UnitMask	Description
7	From Local node to Node 7.
6	From Local node to Node 6.
5	From Local node to Node 5.
4	From Local node to Node 4.
3	From Local node to Node 3.
2	From Local node to Node 2.
1	From Local node to Node 1.
0	From Local node to Node 0.

NBPMCx1E2 CPU Read Command Latency to Target Node 0-3

This event counts the number of NB clocks from when the targeted command is received in the NB to when the targeted command completes. This event only tracks one outstanding command at a time. To determine latency between the local node and a remote node set UnitMask[7:4] to select the node and UnitMask[3:0] to select the read command type. The count returned by the counter should be divided by the count returned by NBPMCx1E3 do determine the average latency for the command type.

UnitMask	Description
7	From Local node to Node 3.

6	From Local node to Node 2.
5	From Local node to Node 1.
4	From Local node to Node 0.
3	Change-to-Dirty.
2	Read block modified.
1	Read block shared.
0	Read block.

NBPMCx1E3 CPU Read Command Requests to Target Node 0-3

This event counts the number of requests that a latency measurement is made for using NBPMCx1E2. To determine the number of commands that a latency measurement are made for between the local node and a remote node set UnitMask[7:4] to select the node and UnitMask[3:0] to select the read command type.

UnitMask	Description
7	From Local node to Node 3.
6	From Local node to Node 2.
5	From Local node to Node 1.
4	From Local node to Node 0.
3	Change-to-Dirty.
2	Read block modified.
1	Read block shared.
0	Read block.

NBPMCx1E4 CPU Read Command Latency to Target Node 4-7

This event counts the number of NB clocks from when the targeted command is received in the NB to when the targeted command completes. This event only tracks one outstanding command at a time. To determine latency between the local node and a remote node set UnitMask[7:4] to select the node and UnitMask[3:0] to select the read command type. The count returned by the counter should be divided by the count returned by NBPMCx1E5 do determine the average latency for the command type.

UnitMask	Description
7	From Local node to Node 7.
6	From Local node to Node 6.
5	From Local node to Node 5.
4	From Local node to Node 4.
3	Change-to-Dirty.
2	Read block modified.
1	Read block shared.
0	Read block.

NBPMCx1E5 CPU Read Command Requests to Target Node 4-7

This event counts the number of requests that a latency measurement is made for using NBPMCx1E4. To determine the number of commands that a latency measurement are made for between the local node and a remote node set UnitMask[7:4] to select the node and UnitMask[3:0] to select the read command type.

UnitMask	Description
7	From Local node to Node 7.
6	From Local node to Node 6.
5	From Local node to Node 5.
4	From Local node to Node 4.
3	Change-to-Dirty.
2	Read block modified.
1	Read block shared.
0	Read block.

NBPMCx1E6 CPU Command Latency to Target Node 0-3/4-7

This event counts the number of NB clocks from when the targeted command is received in the NB to when the targeted command completes. This event only tracks one outstanding command at a time. To determine latency between the local node and a remote node set UnitMask[7:4] to select the node, UnitMask[3] to select the node group and UnitMask[3:0] to select the command type. The count returned by the counter should be divided by the count returned by NBPMCx1E7 do determine the average latency for the command type.

UnitMask	Description
7	From Local node to Node 3/7.
6	From Local node to Node 2/6.
5	From Local node to Node 1/5.
4	From Local node to Node 0/4.
3	Node Group Select. 0=Nodes 0-3. 1= Nodes 4-7.
2	Victim Block.
1	Write Sized.
0	Read Sized.

NBPMCx1E7 CPU Requests to Target Node 0-3/4-7

This event counts the number of requests that a latency measurement is made for using NBPMCx1E6. To determine the number of commands that a latency measurement are made for between the local node and a remote node set UnitMask[7:4] to select the node, UnitMask[3] to select the node group and UnitMask[3:0] to select the command type.

UnitMask	Description
7	From Local node to Node 3/7.
6	From Local node to Node 2/6.
5	From Local node to Node 1/5.

4	From Local node to Node 0/4.
3	Node Group Select. 0=Nodes 0-3. 1= Nodes 4-7.
2	Victim Block.
1	Write Sized.
0	Read Sized.

NBPMCx1EB Request Cache Status 1

The probe response type for RdBlkM or ChgToDirty request type.

UnitMask	Description
7	Track Cache Stat for RdBlkM.
6	Track Cache Stat for ChgToDirty.
5	Directed Probe.
4	Probe Miss.
3	Probe Hit M.
2	Probe Hit MuW or O.
1	Probe Hit E.
0	Probe Hit S.

3.25.5 NBPMCx1F[F:0] Events (Memory Controller, Crossbar)

NBPMCx1F0 Memory Controller Requests

This event counts the specified requests handled by the memory controller.

UnitMask	Description
7	Read Requests while Writes Pending in DCQ.
6	64-Byte Sized Reads from CPU/Hostbridge (excluding internal GPU).
5	32-Byte Sized Reads from CPU/Hostbridge (excluding internal GPU).
4	64-Byte Sized Writes from CPU/Hostbridge (excluding internal GPU).
3	32-Byte Sized Writes from CPU/Hostbridge (excluding internal GPU).
2	Prefetch Requests.
1	Read Requests from CPU/Hostbridge (including Prefetch) and internal GPU.
0	Write Requests from CPU/Hostbridge and internal GPU.

NBPMCx3EC DRAM Accesses

The number of memory accesses performed by the local DRAM controller. UnitMask[7:0] may be used to isolate the different DRAM page access cases. Page miss cases incur an extra latency to open a page; page conflict cases incur both a page-close as well as page-open penalties. These penalties may be overlapped by DRAM accesses for other requests and don't necessarily represent lost DRAM bandwidth. The associated penalties are as follows:

Page miss: Trcd (DRAM RAS-to-CAS delay)

Page conflict: Trp + Trcd (DRAM row-precharge time plus RAS-to-CAS delay)

Each DRAM access represents one 64-byte block of data transferred if the DRAM is configured for 64-byte granularity, or one 32-byte block if the DRAM is configured for 32-byte granularity. (The latter is only applicable to single-channel DRAM systems, which may be configured either way.)

UnitMask	Description
7:3	Reserved.
2	DCT Page Conflict.
1	DCT Page Miss.
0	DCT Page hit.

NBPMCx3ED DRAM Controller Page Table Overflows

The number of page table overflows in the local DRAM controller. This table maintains information about which DRAM pages are open. An overflow occurs when a request for a new page arrives when the maximum number of pages are already open. Each occurrence reflects an access latency penalty equivalent to a page conflict.

UnitMask	Description
7:1	Reserved.
0	DCT Page Table Overflow.

NBPMCx3EE Memory Controller DRAM Command Slots Missed

UnitMask	Description
7:1	Reserved.
0	DCT Command Slots Missed (in MEMCLKs).

NBPMCx3EF Memory Controller Turnarounds

The number of turnarounds on the local DRAM data bus. UnitMask[7:0] may be used to isolate the different cases. These represent lost DRAM bandwidth, which may be calculated as follows (in bytes per occurrence):

DIMM turnaround: DRAM_width_in_bytes * 2 edges_per_memclk * 2
R/W turnaround: DRAM_width_in_bytes * 2 edges_per_memclk * 1
R/W turnaround: DRAM_width_in_bytes * 2 edges_per_memclk * (Tcl-1)

where DRAM_width_in_bytes is 8 or 16 (for single- or dual-channel systems), and Tcl is the CAS latency of the DRAM in memory system clock cycles (where the memory clock for DDR-400, or PC3200 DIMMS, for example, would be 200 MHz).

UnitMask	Description
7:3	Reserved.
2	DCT Write to read turnaround.

1	DCT Read to write turnaround.
0	DCT DIMM (chip select) turnaround.

3.26 FCH Registers

3.26.1 Legacy Block Configuration Registers (IO)

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

IO000 Dma_Ch 0

Bits	Description
15:0	**DmaCh0**. Read-write. Reset: 0. DMA1 channel 0 base and current address.

IO002 Dma_Ch 1

Bits	Description
15:0	**DmaCh1**. Read-write. Reset: 0. DMA1 channel 1 base and current address.

IO004 Dma_Ch 2

Bits	Description
15:0	**DmaCh2**. Read-write. Reset: 0. DMA1 channel 2 base and current address.

IO006 Dma_Ch 3

Bits	Description
15:0	**DmaCh3**. Read-write. Reset: 0. DMA1 channel 3 base and current address.

IO008 Dma_Status

Bits	Description
7:0	**DmaStatus**. Read-write. Reset: 0. Specifies the DMA status register for channels 0-3 for reads; specifies the DMA control register for channels 0-3 for writes.

IO009 Dma_WriteRequest

Bits	Description
7:0	**DmaWriteRequest**. Read-write. Reset: 0. Request register.

IO00A Dma_WriteMask

Bits	Description
7:0	**DmaWriteMask**. Read-write. Reset: 0. DMA channel mask register.

IO00B Dma_WriteMode

Bits	Description
7:0	**DmaWriteMode**. Read-write. Reset: 0. DMA mode register.

IO00C Dma_Clear

Bits	Description
7:0	**DmaClear**. Read-write. Reset: 0. Channel 0-3 DMA clear byte pointer.

IO00D Dma_MasterClr

Dma_MasterClr register

Bits	Description
7:0	**DmaMasterClr**. Read-write. Reset: 0. Write: Channel 0-3 master clear register. Read: Intermediate register.

IO00E Dma_ClrMask

Bits	Description
7:0	**DmaClrmask**. Read-write. Reset: 0. Channel 0-3 DMA clear mask.

IO00F Dma_AllMask

Bits	Description
7:0	**DmaAllMask**. Read-write. Reset: 0. General mask register.

IO020 IntrCntrl1Reg1

Bits	Description
7:0	**IntrCntrl1Reg1**. Read-write. Reset: 0. IRQ0-7 status and control. Read: IRR, ISR. Write: ICW1, OCW2, OCW3.

IO021 IntrCntrl1Reg2

Bits	Description
7:0	**IntrCntrl1Reg2**. Read-write. Reset: 0. IRQ0-7 status and control. Read: IMR. Write: ICW2, ICW3, ICW4, OCW1.

IO022 IMCR_Index

Bits	Description
7:0	**ImcrIndex**. Read-write. Reset: 0. Index port for IMCR register. See IO023 [IMCR_Data].

IO023 IMCR_Data

Bits	Description
7:0	**ImcrData**. Read-write. Reset: 0. Data port for IMCR register. The actual IMCR register is located at index 70h and it is at bit 0.

IO040 TimerCh0

Bits	Description
7:0	**TimerCh0**. Read-write. Reset: 0. 8254 Timer 1 Counter 0 Data Port.

IO041 TimerCh1

Bits	Description
7:0	**TimerCh1**. Read-write. Reset: 0. 8254 Timer 1 Counter 1 Data Port.

IO042 TimerCh2

Bits	Description
7:0	**TimerCh2**. Read-write. Reset: 0. 8254 Timer 1 Counter 2 Data Port.

IO043 Tmr1CntrlWord

Bits	Description
7:6	**CounterSelect**. Read-write. Reset: 0. Bits — Definition 00b — Select counter 0 01b — Select counter 1 10b — Select counter 2 11b — Read back command
5:4	**CommandSelect**. Read-write. Reset: 0. Bits — Definition 00b — Counter latch command 01b — Read/write least significant byte 10b — Read/write most significant byte 11b — Read/write least, and then most significant byte

3:1	**ModeSelect**. Read-write. Reset: 0. Bits — Definition 000b — Asserts OUT signal at end of count 001b — Hardware re-triggerable one-shot 010b — Rate generator 011b — Square wave output 100b — Software triggered strobe 101b — Hardware triggered strobe 111b-110b — Reserved
0	**CntDownSelect**. Read-write. Reset: 0. 0=Binary countdown. 1=BCD countdown.

IO060 Keyboard Data

Bits	Description
7:0	**Data**. Read-write. Reset: 0. Specifies the keyboard controller data port. It should only be read from when IO064[0] reads back 1. It should only be written to when IO064[1] reads back 0.

IO061 Nmi Status

Bits	Description
7	**ParityErrNmi**. Read-only. Reset: X. NMI is caused by parity error (either PERR# or SERR#).
6	**IoChkNmi**. Read-only. Reset: X. NMI is triggered by serial IOCHK.
5	**SpkrClk**. Read-only. Reset: X. The output of 8254 timer counter 2.
4	**RefClk**. Read-only. Reset: X. The output of 8254 timer counter 1.
3	**IoChkNmiEn**. Read-write. Reset: 1. 0=Enable IoChk to NMI generation. 1=Disable IoChk to NMI generation.
2	**ParityErrNmiEn**. Read-write. Reset: 1. 0=Enable Parity Error to NMI generation (from SERR# or PERR#). 1=Disable Parity Error to NMI generation and clear bit[7].
1	**SpkrTmrEnable**. Read-write. Reset: 0. 0=Speaker timer off. 1=Speaker timer on.
0	**SpkrEnable**. Read-write. Reset: 0. 0=Disable counter 2. 1=Enable counter 2.

IO064 Keyboard Command Status

This register reads the keyboard status on CPU reads and writes specify the keyboard command.

Bits	Description
7	**KeyPar**. Read-write; Updated-by-hardware. Reset: 0. 1=Keyboard data transmission parity error.
6	**KeyRTO**. Read-write; Updated-by-hardware. Reset: 0. 1= Receive time out.
5	**KeyTTO**. Read-write; Updated-by-hardware. Reset: 0. 1=Keyboard transmit time out or mouse data available.
4	**KeyEnB**. Read-write; Updated-by-hardware. Reset: 0. 0=Password protected.
3	**CmdData**. Read-write; Updated-by-hardware. Reset: 0. 1=Between the keyboard command and data ports, the command port at 64h was last written. 0=Between the keyboard command and data ports, the data port at 60h was last written.

2	**KeySys**. Read-write; Updated-by-hardware. Reset: 0. Specifies system flag status. 1=Power up or reset. 0=Soft reset.
1	**KeyIBF**. Read-write; Updated-by-hardware. Reset: 0. 1=Input buffer full. 0=Ready to receive next command/data.
0	**KeyOBF**. Read-write; Updated-by-hardware. Reset: 0. 1=Data is available to be read back. 0=No data available.

IO070 RtcAddrPort and NmiMask

Bits	Description
7	**NmiMask**. Write-only. Reset: 0. 0=NMI enabled. 1=NMI masked.
6:0	**RtcAddrPort: RTC Address Port**. Read-write. Reset: 0. This is used with either the internal RTC or an external RTC. This port specifies the index to access the RTC time registers and the CMOS RAM space. IO073_x0A[DV0] should be programmed first to select bank when accessing the CMOS RAM space. See IO073_x00.

IO071 RtcDataPort

Bits	Description
7:0	**RtcDataPort: RTC Data Port**. Read-write. Reset: 0. This is used with either the internal RTC or an external RTC in conjunction with IO070.

IO072 Alternate RTC AddrPort

Bits	Description
7:0	**AlternatRTCAddrPort**. Read-write. Reset: 0. This is used with the internal RTC. This port allows the user to specify the full 8-bit address (instead of bank0/bank1 indexing) to access the 256 bytes of RTC RAM.

IO073 Alternate RTC DataPort

Bits	Description
7:0	**AlternatRTCDataPort**. Read-write. Reset: 0. This is used with the internal RTC in conjunction with IO072.

IO073_x00 RTC Seconds

Bits	Description
7:0	**Seconds**. Read-write. Reset: 0. BCD format. The range of values for this register is 00 through 59. If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by the RTC logic once per second.

IO073_x01 RTC Seconds Alarm

Bits	Description
7:0	**SecondsAlarm: Seconds Alarm**. Read-write. Reset: 0. BCD format. If (IO073_x0B[SET] == 1), the Seconds Alarm register never matches IO073_x00 [RTC Seconds]. Else, if (bits[7:6] == 11b) the RTC Seconds Alarm register always matches with RTC Seconds register and causes an RTC alarm event to be generated once per second. See IO073_x0C[AF].

IO073_x02 RTC Minutes

Bits	Description
7:0	**Minutes**. Read-write. Reset: 0. BCD format. The range of values for this register is 00 through 59. If (IO073_x0B[SET] == 1), this register programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by the RTC logic once per second.

IO073_x03 RTC Minutes Alarm

Bits	Description
7:0	**MinutesAlarm: Minutes Alarm**. Read-write. Reset: 0. BCD format. If (IO073_x0B[SET] == 1), the RTC Minutes Alarm register never matches with IO073_x02 [RTC Minutes]. Else, if (bits[7:6] == 11b), the RTC Minutes Alarm register always matches with RTC Minutes register and causes an RTC alarm event to be generated once per minute. See IO073_x0C[AF].

IO073_x04 RTC Hours

Bits	Description
7:0	**Hours**. Read-write. Reset: 0. BCD format. The range of values for this register is 00 through 23. If (IO073_x0B[SET] == 1), this register can be set by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by the RTC logic once per second.

IO073_x05 RTC Hours Alarm

Bits	Description
7:0	**HoursAlarm: Hours Alarm**. Read-write. Reset: 0. BCD format. If (IO073_x0B[SET] == 1), the RTC Hours Alarm register never matches with IO073_x04 [RTC Hours]. Else, if (bits[7:6] == 11b) the RTC Hours Alarm register always matches with the RTC Hours register and causes an RTC alarm event to be generated once per hour. See IO073_x0C[AF].

IO073_x06 RTC Day of Week

Bits	Description
7:0	**DayOfWeek: Day of Week**. Read-write. Reset: 0. BCD format. The range of values for this register is 1 through 7 (Sunday = 1). Leap year correction is performed by software. If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by hardware.

IO073_x07 RTC Date of Month

Bits	Description
7:0	**DateOfMonth: Date of Month**. Read-write. Reset: 0. BCD format. The range for this register is 01 through 31. Leap year correction is performed by software. If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by hardware.

IO073_x08 RTC Month

Bits	Description
7:0	**Month**. Read-write. Reset: 0. BCD format. The range for this register is 01 through 12. Leap year correction is performed by software. If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by hardware.

IO073_x09 RTC Year

Bits	Description
7:0	**Year**. Read-write. Reset: 0. BCD format. The range for this register is 00 through 99. No leap year correction capability. Leap year correction is done by software. If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is updated by hardware.

IO073_x0A RTC Register A

Bits	Description
7	**UIP: Update In Progress**. Read-only. Reset: 0. If (IO073_x0B[SET] == 1), UIP is cleared. 1=The update transfer is imminent. 0=The update transfer does not occur for at least 244 us.
6:5	Reserved.

4	**DV0: Bank Selection**. Read-write. Reset: 0. 0=Select bank 0 when accessing the RTC CMOS RAM space through IO070 and IO071. 1=Select bank 1 when accessing the RTC CMOS RAM space through IO070 and IO071. The FCH has an alternate way to access the RAM without the use of bank select bit. IO072 [Alternate RTC AddrPort] and IO073 [Alternate RTC DataPort] provides indexed access to the full 256 bytes of RAM.
3:0	**RS[3:0]: Rate Selection**. Read-write. Reset: 0. Specifies one of 13 taps on the 15-stage frequency divider or disables the divider output when RS[3:0] == 0. The tap selected can be used to generate a periodic interrupt. See the following table for the frequency selection: Bits — Tap Frequency (Interrupt Rate) 0000b — Flat Signal (None) 0001b — 256 Hz (3.90625 ms) 0010b — 128 Hz (7.8125 ms) 0011b — 8.192 kHz (122.070 us) 0100b — 4.096 kHz (244.141 us) 0101b — 2.048 kHz (488.281 us) 0110b — 1.024 kHz (976.5625 us) 0111b — 512 Hz (1.953125 ms) 1000b — 256 Hz (3.90625 ms) 1001b — 128 Hz (7.8125 ms) 1010b — 64 Hz (15.625 ms) 1011b — 32 Hz (31.25 ms) 1100b — 16 Hz (62.5 ms) 1101b — 8 Hz (125 ms) 1110b — 4 Hz (250 ms) 1111b — 2 Hz (500 ms)

IO073_x0B RTC Register B

Bits	Description
7	**SET: Set new time**. Read-write. Reset: 0. 1=No internal updating for RTC time registers occurs. 0=The RTC time registers are updated every second.
6	**PIE: Periodic Interrupt Enable**. Read-write. Reset: 0. 1=Enable the IO073_x0C[PF] bit to assert IRQ.
5	**AIE: Alarm Interrupt Enable**. Read-write. Reset: 0. 1=Enable the IO073_x0C[AF] bit to assert IRQ. When the alarm time is written in the appropriate hours, minutes, and seconds alarm registers, the alarm interrupt is initiated at the specified time each day if AIE == 1.
4	**UIE: Update Ended Interrupt Enable**. Read-write. Reset: 0. 1=Enable the IO073_x0C[UF] bit to assert IRQ. If [SET] == 1, UIE is cleared.
3:2	Reserved.
1	**HourMode**. Read-write. Reset: 0. 0=12 hour mode. 1=24 hour mode.
0	**DaylightSavingEnable**. Read-write. Reset: X. BIOS: 0. 1=If (PMx5F_x00[DltSavEnable] == 1) RTC daylight saving is enabled.

IO073_x0C RTC Register C

Bits	Description
7	**IRQF: Interrupt Request Flag**. Read-only. Reset: 0. IRQF=(PF*PIE)+(AF*AIE)+(UF*UIE). 1=The IRQ# pin is driven low. Reading RTC Register C clears IRQF bit.
6	**PF: Periodic Interrupt Flag**. Read-only. Reset: 0. 1=An edge is detected on the selected tap (through RS3 to RS0) of the frequency divider. Reading RTC Register C clears PF bit.
5	**AF: Alarm Interrupt Flag**. Read-only. Reset: 0. 1=IO073_x00 [RTC Seconds], IO073_x02 [RTC Minutes], IO073_x04 [RTC Hours] and IO073_x07 [RTC Date of Month] match IO073_x01 [RTC Seconds Alarm], IO073_x03 [RTC Minutes Alarm], IO073_x05 [RTC Hours Alarm], and IO073_x0D [RTC Date Alarm] respectively. Reading RTC Register C clears AF bit.
4	**UF: Update Ended Interrupt Flag**. Read-only. Reset: 0. 1=Update cycle complete. Reading RTC Register C clears UF.
3:0	Reserved.

IO073_x0D RTC Date Alarm

Bits	Description
7	**VRT: Valid RAM and Time**. Read-only. Reset: 1. 1=RTC date, time, and CMOS RAM are valid. 0=RTC date, time, and CMOS RAM are invalid due to low RTC battery being monitored. See PMx58 [VRT_T1] and PMx59 [VRT_T2].
6	**ScratchBit**. Read-write. Reset: 0.
5:0	**DateAlarm**. Read-write. Reset: 0. If (DateAlarm != 0), then DateAlarm is considered for alarm generation. 0=DateAlarm is not compared for alarm generation. DateAlarm is in BCD format.

IO073_x32 RTC AltCentury

Bits	Description
7:0	**AltCentury**. Read-write. Reset: 0. BCD format. Leap year correction is performed by hardware. This register is accessed only when (IO073_x0A[DV0] == 0) and (PMx56[CenturyEn] == 1). If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is automatically updated by hardware every century.

IO073_x48 RTC Century

Bits	Description
7:0	**Century**. Read-write. Reset: 0. BCD format. Leap year correction is done through hardware. This register is accessed only when (IO073_x0A[DV0] == 1). If (IO073_x0B[SET] == 1), this register is programmed by software and hardware updating is disabled. If (IO073_x0B[SET] == 0), this register is automatically updated by hardware every century.

IO073_x50 RTC Extended RAM Address Port

Bits	Description
7	Reserved.
6:0	**ExtendedRamAddrPort**. Read-write. Reset: 0. Because only 7 address bits are used in IO070[RtcAddrPort], only the lower 128 bytes at offset 7Fh:00h are accessible through IO071 [RtcDataPort]. The Extended RAM is physically located at address 80h to FFh. In order to access these addresses, an address offset should be programmed into this register and access to them is done through IO073_x53 [RTC Extended RAM Data Port]. An offset of 80h is automatically added to this 7-bit address.

IO073_x53 RTC Extended RAM Data Port

Bits	Description
7:0	**ExtendedRamDataPort**. Read-write. Reset: X.

IO073_x7E RTC Time Clear

Bits	Description
7:1	Reserved.
0	**RtcTimeClear**. Read-write. Reset: 0. 1=Clear the RTC second and stop RTC time.

IO073_x7F RTC RAM Enable

Bits	Description
7:1	Reserved.
0	**RtcRamEnable**. Read-write. Reset: 1. 1=Enable access to the RTC RAM.

IO080 PostCode 0

Bits	Description
7:0	**PostCode[7:0]: BIOS post code**. Read-write. Reset: 0. BIOS post code register, can be up to 4 bytes for writes and 1 byte for reads. See MISCx6C[PostCodeWidthSel], D14F3x48[IOPortEnable4] and MISCx78 [PostCode].

IF (!MISCx6C[PostCodeWidthSel] && D14F3x48[IOPortEnable4]) THEN

IO081 PostCode 1

Bits	Description
7:0	**PostCode[15:8]: BIOS post code**. Read-write. Reset: 0.

ELSE

IO081 Dma_PageCh2

Bits	Description
7:0	**DmaPagech2**. Read-write. Reset: 0. DMA2 channel 2 page register.

ENDIF.
IF (!MISCx6C[PostCodeWidthSel] && D14F3x48[IOPortEnable4]) THEN

IO082 PostCode 2

Bits	Description
7:0	**PostCode[23:16]: BIOS post code**. Read-write. Reset: 0.

ELSE

IO082 Dma_PageCh3

Bits	Description
7:0	**DmaPageCh3**. Read-write. Reset: 0. DMA2 channel 3 page register.

ENDIF.
IF (!MISCx6C[PostCodeWidthSel] && D14F3x48[IOPortEnable4]) THEN

IO083 PostCode 3

Bits	Description
7:0	**PostCode[31:24]: BIOS post code**. Read-write. Reset: 0.

ELSE

IO083 Dma_PageCh1

Bits	Description
7:0	**DmaPageCh1**. Read-write. Reset: 0. DMA2 channel 1 page register.

ENDIF.

IO084 Dma_Page_Reserved1

Bits	Description
7:0	**DmaPageReserved1**. Read-write. Reset: 0. DMA Page Reserved1 register.

IO085 Dma_Page_Reserved2

Bits	Description
7:0	**DmaPageReserved2**. Read-write. Reset: 0. DMA Page Reserved2 register.

IO086 Dma_Page_Reserved3

Bits	Description
7:0	**DmaPageReserved3**. Read-write. Reset: 0. DMA Page Reserved3 register.

IO087 Dma_PageCh0

Bits	Description
7:0	**DmaPageCh0**. Read-write. Reset: 0. DMA2 channel 0 page register.

IO088 Dma_Page_Reserved4

Bits	Description
7:0	**DmaPageReserved4**. Read-write. Reset: 0. DMA Page Reserved4 register.

IO089 Dma_PageCh6

Bits	Description
7:0	**DmaPageCh6**. Read-write. Reset: 0. DMA2 channel 6 page register.

IO08A Dma_PageCh7

Bits	Description
7:0	**DmaPageCh7**. Read-write. Reset: 0. DMA2 channel 7 page register.

IO08B Dma_PageCh5

Bits	Description
7:0	**DmaPageCh5**. Read-write. Reset: 0. DMA2 channel 5 page register.

IO08C Dma_Page_Reserved5

Bits	Description
7:0	**DmaPageReserved5**. Read-write. Reset: 0. DMA Page Reserved5 register.

IO08D Dma_Page_Reserved6

Bits	Description
7:0	**DmaPageReserved6**. Read-write. Reset: 0. DMA Page Reserved6 register.

IO08E Dma_Page_Reserved7

Bits	Description
7:0	**DmaPageReserved7**. Read-write. Reset: 0. DMA Page Reserved7 register.

IO08F Dma_Refresh

Bits	Description
7:0	**DmaRefresh**. Read-write. Reset: 0. DMA2 channel 4 page register.

IO092 FastInit

Bits	Description
7:2	Reserved.
1	**A20EnB: A20 Enable Bar bit**. Read-write. Reset: 0. 1=A20M# function is disabled.
0	**FastInit**. Read-write. Reset: 0. This bit provides a fast software executed processor reset function. 1=Generate INIT assertion for approximately 4 ms. 0=Before another INIT pulse can be generated via this register, this bit must be programmed back to 0.

IO0A0 IntrCntrl2Reg1

Bits	Description
7:0	**IntrCntrl2Reg1**. Read-write. Reset: 0. IRQ8-15 status and control. Read: IRR, ISR. Write: ICW1, OCW2, OCW3.

IO0A1 IntrCntrl2Reg2

Bits	Description
7:0	**IntrCntrl2Reg2**. Read-write. Reset: 0. IRQ8-15 status and control. Read: IMR. Write: ICW2, ICW3, ICW4, OCW1.

IO0C0 Dma2_Ch4Addr

Bits	Description
7:0	**Dma2Ch4Addr**. Read-write. Reset: 0. DMA2 channel 4 base and current address.

IO0C2 Dma2_Ch4Cnt

Bits	Description
7:0	**Dma2Ch4Cnt**. Read-write. Reset: 0. DMA2 channel 4 base and current count.

IO0C4 Dma2_Ch5Addr

Bits	Description
7:0	**Dma2Ch5Addr**. Read-write. Reset: 0. DMA2 channel 5 base and current address.

IO0C6 Dma2_Ch5Cnt

Bits	Description
7:0	**Dma2Ch5Cnt**. Read-write. Reset: 0. DMA2 channel 5 base and current count.

IO0C8 Dma2_Ch6Addr

Bits	Description
7:0	**Dma2Ch6Addr**. Read-write. Reset: 0. DMA2 channel 6 base and current address.

IO0CA Dma2_Ch6Cnt

Bits	Description
7:0	**Dma2Ch6Cnt**. Read-write. Reset: 0. DMA2 channel 6 base and current count.

IO0CC Dma2_Ch7Addr

Bits	Description
7:0	**Dma2Ch7Addr**. Read-write. Reset: 0. DMA2 channel 7 base and current address.

IO0CE Dma_Ch7Cnt

Bits	Description
7:0	**Dma2Ch7Cnt**. Read-write. Reset: 0. DMA2 channel 7 base and current count.

IO0D0 Dma_Status

Bits	Description
7:0	**DmaStatus**. Read-write. Reset: 0. DMA2 status register.

IO0D2 Dma_WriteRequest

Bits	Description
7:0	**DmaWriteRequest**. Read-write. Reset: 0. DMA2 request register.

IO0D4 Dma_WriteMask

Bits	Description
7:0	**DmaWriteMask**. Read-write. Reset: 0. DMA2 channel mask register.

IO0D6 Dma_WriteMode

Bits	Description
7:0	**DmaWriteMode**. Read-write. Reset: 0. DMA2 mode register.

IO0D8 Dma_Clear

Bits	Description
7:0	**DmaClear**. Read-write. Reset: 0. Channel 4-7 clear byte pointer.

IO0DA Dma_MasterClr

Bits	Description
7:0	**DmaMasterClear**. Read-write. Reset: 0. Write: Channel 4-7 DMA master clear. Read: Intermediate register.

IO0DC Dma_ClrMask

Bits	Description
7:0	**DmaClrMask**. Read-write. Reset: 0. Channel 4-7 DMA clear mask.

IO0DE Dma_AllMask

Bits	Description
7:0	**DmaAllMask**. Read-write. Reset: 0. DMA2 mask register.

IO0F0 NCP Error

Writes to this port assert IGNNE# if FERR# is true. If FERR# is false, writes to this port do not assert IGNNE#. The first write sets WarmBoot.

Bits	Description
7	**WarmBoot**. Read; write-once. Cold reset: 0. Warm or cold boot indicator. 0=Cold. 1=Warm.
6:0	Reserved.

IO4D0 IntrEdgeControl

Bits	Description
15	**IRQ15Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
14	**IRQ14Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
13	Reserved.
12	**IRQ12Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
11	**IRQ11Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
10	**IRQ10Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
9	**IRQ9Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
8	**IRQ8Control**. Read-only. Reset: 0. Always edge.
7	**IRQ7Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
6	**IRQ6Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
5	**IRQ5Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
4	**IRQ4Control**. Read-write. Reset: 0. 1=Level. 0=Edge.
3	**IRQ3Control.** Read-write. Reset: 0. 1=Level. 0=Edge.
2	Reserved.
1	**IRQ1Control.** Read-write. Reset: 0. 1=Level. 0=Edge.
0	**IRQ0Control.** Read-write. Reset: 0. 1=Level. 0=Edge.

IOC00 Pci_Intr_Index

Bits	Description
7	**PciIntrApic**. Read-write. Reset: 0. 0=IRQ routing to PIC. 1=IRQ routing to IOAPIC.
6:0	**PciIntrIndex**. Read-write. Reset: 0. PCI interrupt index. Selects which PCI interrupt to map.

Bits	Definition	Bits	Definition
00h	INTA#	1Ah	SDIO
01h	INTB#	1Fh-1Bh	Reserved
02h	INTC#	20h	IMC INT0
03h	INTD#	21h	IMC INT1
04h	GENINT1/INTE#	22h	IMC INT2
05h	GENINT2/INTF#	23h	IMC INT3
06h	INTG#	24h	IMC INT4
07h	INTH#	25h	IMC INT5
08h	Misc	2Fh-26h	Reserved
09h	Misc0	30h	Dev12h(USB) INTA#
0Ah	Misc1	31h	Reserved
0Bh	Misc2	32h	Dev13h(USB) INTA#
0Ch	INTA from serial IRQ	33h	Reserved
0Dh	INTB from serial IRQ	34h	Dev16h(USB) INTA#, Dev10h xHCI
0Eh	INTC from serial IRQ	35h	Reserved
0Fh	INTD from serial IRQ	40h-36h	Reserved
10h	SCI	41h	SATA PCI interrupt
11h	SMBUS0	61h-42h	Reserved
12h	ASF	62h	GPIO controller interrupt
13h	HD audio	6Fh-63h	Reserved
14h	FC	70h	I^2C0
15h	Reserved	71h	I^2C1
16h	PerMon	72h	I^2C2
17h	SD	73h	I^2C3
18h	Reserved	74h	UART0
19h	Reserved	75h	UART1
		7Fh-76h	Reserved

IOC01 Pci_Intr_Data

Bits	Description
7:0	**PciIntrData**. Read-write. Reset: 0.

IOC01_x0[7:0] PCI INT[H#,G#,F#,E#,D#,C#,B#,A#] Map

Bits	Description
7:5	Reserved.
4:0	**Pci2IntrMap**. Read-write. Reset: 1Fh. If (IOC00[PciIntrApic] == 1) then Pci2IntrMap specifies mapping of INT[H#:A#] to APIC interrupt number. If (IOC00[PciIntrApic] == 0) then Pci2IntrMap specifies mapping of INT[H#:A#] to PIC interrupt number.

IOC01_x08 Intr Misc Map

Bits	Description
7:6	**Pci2Intr15Map**. Read-write. Reset: 0. Bits: Definition 00b: IRQ15 mapped to legacy IDE 01b: IRQ15 mapped to SATA IDE 10b: IRQ15 mapped to SATA2 11b: IRQ15 come from Serial IRQ or PCI interrupt
5:4	**Pci2Intr14Map**. Read-write. Reset: 0. Bits: Definition 00b: IRQ14 mapped to legacy IDE 01b: IRQ14 mapped to SATA IDE 10b: IRQ14 mapped to SATA2 11b: IRQ14 mapped to Serial IRQ or PCI interrupt
3	**PciIntrIrq12**. Read-write. Reset: 0. 0=Integrated Micro-Controller (IMC) as IRQ12 input source. 1=Serial IRQ or PCI devices as IRQ12 input source.
2	**PciIntrIrq8**. Read-write. Reset: 0. 0=Real-Time Clock (RTC) is IRQ8 input source. 1=Serial IRQ or PCI devices as IRQ8 input source.
1	**PciIntrIrq1**. Read-write. Reset: 0. 0=IMC as IRQ1 input source. 1=Serial IRQ or PCI devices as IRQ1 input source.
0	**PciIntrIrq0**. Read-write. Reset: 0. 0=8254 timer as IRQ0 input source. 1=Serial IRQ or PCI devices as IRQ0 input source.

IOC01_x09 Intr Misc 0 Map

Bits	Description
7	**IntrDelay**. Read-write. Reset: 1. INTR 600 ns delay.
6	**IRQ12FilterEnable**. Read-write. Reset: 1. IRQ12 filter enable.
5	**IRQ1FilterEnable**. Read-write. Reset: 1. IRQ1 filter enable.
4	**IrqInputEn**. Read-write. Reset: 0. 0=Mask off IRQ input. 1=Enable IRQ input.
3	**MaskIrq1Irq12**. Read-write. Reset: 0. 0=Enable IRQ1 and IRQ12. 1=Mask off IRQ1 and IRQ12.
2	**Merge_Ec_irq12**. Read-write. Reset: 1. 0=Route serial IRQ12 to USB IRQ12 input. 1=Route IMC IRQ12 to USB IRQ12 input.
1	**Merge_Ec_irq1**. Read-write. Reset: 1. 0=Route serial IRQ1 to USB IRQ1 input. 1=Route IMC IRQ1 to USB IRQ1 input.
0	**IntMap**. Read-write. Reset: 1. 0=INT0 in IOAPIC comes from IRQ0 in PIC, INT2 in IOAPIC comes from INTR in PIC. 1=INT2 in IOAPIC comes from IRQ0 in PIC, INT0 in IOAPIC comes from INTR in PIC.

IOC01_x0A IntrMisc1Map

Bits	Description
7:0	**HPET**. Read-write. Reset: 0. Writes to this register update the bits in HPETx1[4:0:Step2]0[47:32]; All 3 registers, HPETx100[47:32], HPETx120[47:32], and HPETx140[47:32], are updated at the same time. IOC01_x0A updates the lower 8 bits. IOC01_x0B updates the upper 8 bits.

IOC01_x0B IntrMisc2Map

Bits	Description
7:0	**HPET**. See: IOC01_x0A[HPET].

IOC01_x0[7F:C] PCI Interrupt Map

Bits	Description
7:5	Reserved.
4:0	**Pci2IntrMap.** Read-write. Reset: 1Fh. If (IOC00[PciIntrApic] == 1), then Pci2IntrMap specifies the APIC interrupt number that the corresponding PCI interrupt maps to. If (IOC00[PciIntrApic] == 0), then Pci2IntrMap specifies the PIC interrupt number mapping to the corresponding PCI interrupt. See IOC00 [Pci_Intr_Index] for the PCI interrupt list.

IOC14 Pci_Error

Bits	Description
7:4	Reserved.
3	**PerrNmi**. Read-write. Reset: 1. Enable NMI generation from PERR#. 0=Enable. 1=Disable.
2	**SerrNmi**. Read-write. Reset: 1. Enable NMI generation from SERR#. 0=Enable. 1=Disable.
1	**PerrNmiStatus**. Read-only. Reset: X. 1=NMI generation is enabled and PERR# is asserted due to a PCI data parity error. This bit is cleared by writing 1 to IO061[2].
0	**SerrNmiStatus**. Read-only. Reset: X. 1=NMI generation is enabled and SERR# is asserted due to a PCI error. This bit is cleared by writing 1 to IO061[2].

IOCD0 PM2_Index

Bits	Description
7:0	**Pm2Index: Power management 2 index register.** Read-write. Reset: 0. This register specifies the index of the power management 2 register. See Power Management Block 2 (PM2) Registers.

IOCD1 PM2_Data

Bits	Description
7:0	**Pm2Data: Power management 2 data register**. Read-write. Reset: 0. This register specifies the data read from or written to the power management 2 register pointed by IOCD0. See Power Management Block 2 (PM2) Registers.

IOCD4 BIO SRAMIndex

Bits	Description
7:0	**BiosRamIndex: BIOS RAM index register**. Read-write. Reset: 0. This register specifies the index in the 256-byte BIOS RAM. Data in this RAM is preserved until RSMRST# is asserted or S5 power is lost.

IOCD5 BIO SRAM Data

Bits	Description
7:0	**BiosRamData: BIOS RAM data register**. Read-write. Reset: 0. This register specifies the data read from/written to the BIOS RAM pointed by IOCD4.

IOCD6 PM_Index

Bits	Description
7:0	**PmIndex: Power management index register**. Read-write. Reset: 0. This register specifies the index of the power management register. See 3.26.12 [Power Management (PM) Registers].

IOCD7 PM_Data

Bits	Description
7:0	**PmData: Power management data register**. Read-write. Reset: 0. This register specifies the data read from or written to the power management register pointed by IOCD6. See 3.26.12 [Power Management (PM) Registers].

IOCF9 System Reset Register

This register can be accessed through PMxC5 [CF9 Shadow].

Bits	Description
7:4	Reserved.
3	**FullRst**. Read-write. Reset: 0. 0=Assert reset signals only. 1=Place system in S5 state for 3 to 5 seconds.
2	**RstCmd**. Read-write; Cleared-by-hardware. Reset: 0. 1=Generate reset as specified by FullRst and SysRst.

1	**SysRst**. Read-write. Reset: 0. 0=Send an INIT HyperTransport™ message. 1=Reset as specified by FullRst.
0	Reserved.

3.26.2 AB Configuration Registers

AB Configuration registers are accessed indirectly through ABx00 [AB Index Register] and ABx04 [AB Data Register]. The ABx00/ABx04 register pair is located in IO address space as defined by PMxE0.

ABx00 AB Index Register

Bits	Description
31:29	**RegSpace**. Read-write. Reset: 0. Bits Definition 101b-000b Reserved 110b ABCFG. A-Link Bridge Configuration 111b Reserved Once a valid register block address is written to this field, only valid register block addresses can be written to this field.
28:17	Reserved.
16:0	**RegAddr: Register Address**. Read-write. Reset: 0.

ABx04 AB Data Register

Bits	Description
31:0	**Data: Register Address**. Read-write. Reset: 0.

ABx04_x54 Misc Control 1

Bits	Description
31:25	Reserved.
24	**BlClkGateEn**. Read-write. Reset: 0. BIOS: See 2.17.10.3. 1=Enable gating of B-Link clocks when idle is detected.
23:16	**BlClkGateDelay**. Read-write. Reset: 10h. BIOS: See 2.17.10.3. Specifies the number of B-Link clocks to delay before gating B-Link clocks after idle condition is detected.
15:5	Reserved.
4	**DbgClkGateEn**. Read-write. Reset: 0. BIOS: See 2.17.10.3. 1=Enable gating of B-Link debug clocks.
3	Reserved.
2	**UpSWrByteCntSbgMode**. Read-write. Reset: 0. BIOS: See 2.17.10.4. 1=Upstream 32/64-byte MST_BIF write requests are issued only when all byte enables in the request are 1. This field is mutually exclusive with UpWr16BMode.
1	Reserved.
0	**UpWr16BMode**. Read-write. Reset: 1. BIOS: See 2.17.10.4. Enables upstream writes to be limited to a maximum of 16 bytes per transaction. 0=Disable 16-Byte mode; AB sends DMA write requests with lengths of 16, 32, or 64 bytes. 1=Enable 16-Byte mode; AB only sends DMA write requests with length 16 bytes. This bit should be mutually exclusive with UpSWrByteCntSbgMode. See ABx04_x204[Dma16ByteMode].

ABx04_x58 B-Link RAB Control

Bits	Description
31	**AlMemSDEn**. Read-write. Reset: 0. BIOS: See 2.17.10.2. 1=Enable AB A-Link Memory Shut Down Feature. See ABx04_x10054[27:26] for idle counter settings.
30	Reserved.
29	**BlMemSDEn**. Read-write. Reset: 0. BIOS: See 2.17.10.2. 1=Enable AB B-Link Memory Shut Down Feature. See ABx04_x10054[27:26] for idle counter settings.
28:0	Reserved.

ABx04_x80 B-Link DMA Prefetch Control

Bits	Description
31:22	Reserved.
21:16	**BlPrefMode**. Read-write. Reset: 0. BIOS: 6. Bit — Definition [5:3] — Reserved. [2] — Must be programmed to 1 to set B-Link prefetch mode. [1] — Must be programmed to 1 to set B-Link prefetch mode. [0] — Reserved.
15:8	Reserved.
7:0	**BlPrefEn**. Read-write. Reset: 0. BIOS: 1. Enable B-Link prefetch on a per-device basis. Bit — Definition [7:1] — Reserved. [0] — USB OHCI.

ABx04_x90 BIF Control 0

Bits	Description
31:22	Reserved.
21	**PpNpRfwCplCtrl**. Read-write. Reset: 0. BIOS: 1. RFW completion control for posted pass non-posted.
20:0	Reserved.

ABx04_x94 MSI Control

Bits	Description
31:21	Reserved.
20	**MsiAddrEn**. Read-write. Reset: 0. BIOS: 1. 1=Enable AB to detect MSI sent upstream from FCH and inform the power management controller.
19:0	**MsiAddr[39:20]**. Read-write. Reset: 0. BIOS: See 2.17.10.1. Specifies the MSI Address[39:20].

ABx04_x204 SBG Upstream Control

Bits	Description
31:1	Reserved.
0	**Dma16ByteMode**. Read-write. Reset: 0. Always split DMA write request to be 16 bytes request. 0=All the DMA write requests have the size that AB passes up. 1=Split the DMA write requests to be 16 bytes requests. This field overrides ABx04_x54[UpWr16BMode].

ABx04_x10054 A-Link Arbitration Control and Clock Control

Bits	Description
31:26	Reserved.
25	**DbgClkGateEn**. Read-write. Reset: 0. BIOS: See 2.17.10.3. 1=Enable gating of A-Link debug clocks.
24	**AlClkGateEn**. Read-write. Reset: 0. BIOS: See 2.17.10.3. 1=Enable gating of A-Link clocks when idle is detected.
23:16	**AlClkGateDelay**. Read-write. Reset: 10h. BIOS: See 2.17.10.3. Specifies the number of cycles to delay before gating A-Link clocks after idle condition is detected.
15:12	Reserved.
11:0	**ArbCtl**. Read-write. Reset: 0. BIOS: 7FFh. A-Link Arbiter Control. 1=Enable the A-Link int_arbiter enhancement to allow the A-Link bandwidth to be used more efficiently.

ABx04_x10090 Misc Control 3

Bits	Description
31:17	Reserved.
16	**IOTrapDelayEnable: IO Trap Delay Enable**. Read-write. Reset: 1. BIOS: 1. 1=Enable the IO trap delay.
15:0	Reserved.

3.26.3 SATA Controller

3.26.3.1 Device 11h Function 0 (SATA) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D11F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID**. Read-only. Reset: Product-specific. SATA controller has different device IDs for different drivers. Bits — Definition 7900h — SATA in IDE mode 7901h — SATA in AHCI mode with MS driver 7904h — SATA in AHCI mode with AMD driver
15:0	**VendorID: Vendor ID**. Read-only. Reset: 1022h.

D11F0x04 Status/Command

Bits	Description
31	**DetectedParityError: Detected Parity Error**. Read; Write-1-to-clear. Reset: 0. 1=The SATA controller detects a parity error.
30	**SerrStatus: SERR# Status**. Read; Write-1-to-clear. Reset: 0. 1=The SATA controller detects a PCI address parity error.
29	**ReceivedMasterAbort: Received Master Abort**. Read; Write-1-to-clear. Reset: 0. 1=The SATA controller aborts a PCI bus memory cycle while acting as a PCI master.
28	**ReceivedTargetAbort: Received Target Abort**. Read; Write-1-to-clear. Reset: 0. 1=The SATA controller generated PCI cycle (SATA controller is the PCI master) is aborted by a PCI target.
27	**SignaledTargetAbort: Signaled Target Abort**. Read; Write-1-to-clear. Reset: 0. 1=The SATA controller signals Target Abort.
26:25	**DevselTiming: DEVSEL# Timing**. Value: 1. These bits indicate DEVSEL# timing when performing a positive decode. 1=DEVSEL# is asserted to meet the medium timing.
24	**DataParityError: Data Parity Error**. Read; Write-1-to-clear. Reset: 0. 1=SATA controller detects PERR# being asserted while acting as PCI master, regardless whether PERR# was driven by SATA controller or not.
23	**FastBack2BackCapable: Fast Back-to-Back Capable**. Value: 0.
22	Reserved.
21	**Support66MHz: 66 MHz Support**. Read-only. Reset: 1. 1=66 MHz capable.
20	**CapabilitiesList: Capabilities List**. Read-only. Reset: 1. 1=Capability list supported.
19	**InterruptStatus: Interrupt Status**. Read-only. Reset: 0. This bit reflects the state of the interrupt in the device/function. If (InterruptDisable==0) && (InterruptStatus==1), the device/function's INTx# signal is asserted. Setting InterruptDisable to 1 has no effect on the state of this bit.
18:11	Reserved.

10	**InterruptDisable: Interrupt Disable**. Read-write. Reset: 0. This bit disables the device/function from asserting INTx#. 0=Enable the assertion of the device/function's INTx# signal. 1=Disable the assertion of the device/function's INTx# signal.
9	**FastBack2BackEnable: Fast Back-to-Back Enable**. Value: 0.
8	**SERREnable: SERR# Enable**. Read-write. Reset: 0. 1=The SATA controller asserts SERR# when it detects an address parity error && (PerrDetectionEnable==1). 0=SERR# is not asserted.
7	**WaitCycleEnable: Wait Cycle Enable**. Value: 0.
6	**PerrDetectionEnable: PERR# Detection Enable**. Read-write. Reset: 0. 1=The host controller asserts PERR# when it is the agent receiving data && when it detects a parity error. 0=PERR# is not asserted.
5	**VGAPaletteSnoopEnable: VGA Palette Snoop Enable**. Value: 0.
4	**MemoryWriteandInvalidateEnable: Memory Write and Invalidate Enable**. Value: 0.
3	**SpecialCycleRecognitionEnable: Special Cycle Recognition Enable**. Value: 0.
2	**BusMasterEnable: Bus Master Enable**. Read-write. Reset: 0. 1=Enable the device behaving as a bus master.
1	**MemoryAccessEnable: Memory Access Enable**. Read-write. Reset: 0. 1=Enable the SATA controller to respond to PCI memory space access.
0	**IOAccessEnable: IO Access Enable**. Read-write. Reset: 0. 1=Enable the SATA controller to respond to PCI IO space access.

D11F0x08 Revision ID/Class Code

SATA controller has different values for Subclass Code and ProgramIF[7:0] depending on its operating mode.

Table 271: SATA Controller Subclass Code and ProgramIF Settings

Subclass Code	ProgramIF[7:0]	Controller Type
01	8Fh/8Ah	IDE
06	01	AHCI

Bits	Description
31:24	**ClassCode: Class Code**. Value: 1. This is a Mass-Storage Controller.
23:16	**SubclassCode**. IF (D11F0x40[SubclassCodeWriteEnable]) THEN Read-write. ELSE Read-only. ENDIF. Reset: 06h. BIOS: See 2.17.3.1. See Table 271. 1=IDE Mode. 6=AHCI Mode.
15	**ProgramIF[7]: Master IDE Device**. IF (D11F0x40[SubclassCodeWriteEnable]) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. BIOS: See 2.17.3.1. 1=Master IDE device in IDE mode. See Table 271.
14:12	Reserved.
11	**ProgramIF[3]: Secondary Operating Mode Programmable**. IF ((D11F0x40[SubclassCodeWriteEnable]) \|\| (D11F0x08[SubclassCode] == 01h)) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. BIOS: See 2.17.3.1. 1=Both secondary operating modes are supported in IDE. See Table 271.

10	**ProgramIF[2]: Secondary Operating Mode.** IF ((D11F0x40[SubclassCodeWriteEnable] == 1) \|\| (D11F0x08[SubclassCode] == 01h)) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. BIOS: See 2.17.3.1. See Table 271. Bits Definition 0b Compatibility Mode in IDE 1b Native PCI Mode in IDE
9	**ProgramIF[1]: Primary Operating Mode Programmable**. IF ((D11F0x40[SubclassCodeWriteEnable] == 1) \|\| (D11F0x08[SubclassCode] == 01h)) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. BIOS: See 2.17.3.1. 1=Both Primary operating modes are supported in IDE mode. See Table 271.
8	**ProgramIF[0]: Primary Operating Mode.** IF ((D11F0x40[SubclassCodeWriteEnable] == 1) \|\| (D11F0x08[SubclassCode] == 01h)) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. BIOS: See 2.17.3.1. See Table 271. Bits Definition 0b Compatibility Mode in IDE 1b Native PCI Mode in IDE
7:0	**RevisionID: Revision ID**. Read-only. Reset: 49h.

D11F0x0C Header Type Register

Bits	Description
31	**BISTCapable: BIST Capable**. Value: 0. 0=No HBA related BIST function.
30	**StartBIST: Start BIST**. Read-write. Reset: 0. Programming this bit has no effect.
29:28	Reserved.
27:24	**CompletionCode: Completion Code**. Read-only. Reset: 0. Indicates the completion code status of BIST. A non-zero value indicates a failure.
23	**MultiFunctionDevice: Multi-function device**. Read-only. Reset: 0. 0=Single-function device.
22:16	**HeaderType: Header Type**. Read-only. Reset: 0. Since the SATA host controller is a single-function device, this field contains a value of 00h.
15:11	**MasterLatencyTimer: Master Latency Timer**. Read-write. Reset: 0. Specifies, in units of PCI bus clocks, the time slice allowed to SATA host controller for burst transactions.
10:8	Reserved.
7:4	**CacheLineSize: Cache Line Size**. Read-write. Reset: 0. 1=Cache line size is 16 DW (64 bytes).
3:0	Reserved.

D11F0x10 Primary IDE CS0 Base Address(BAR0)

Bits	Description
31:3	**PrimaryIDECS0BaseAddress: Primary IDE CS0 Base Address**. Read-write. Reset: 0. Base address[31:3] for Primary IDE Bus CS0 in IDE mode. This register is used for native mode only. See 3.26.3.2.1 [IDE Compatibility Mode and Native Mode Registers].
2:1	Reserved.
0	**ResourceTypeIndicator: Resource Type Indicator**. Value: 1. 1=Base address field in this register maps to the IO space.

D11F0x14 Primary IDE CS1 Base Address(BAR1)

Bits	Description
31:2	**PrimaryIDECS1BaseAddress: Primary IDE CS1 Base Address**. Read-write. Reset: 0. Base address[31:2] for Primary IDE Bus CS1 in IDE mode. This register is used for native mode only. See 3.26.3.2.1 [IDE Compatibility Mode and Native Mode Registers].
1	Reserved.
0	**ResourceTypeIndicator: Resource Type Indicator**. Value: 1. 1=Base address field in this register maps to the IO space.

D11F0x18 Secondary IDE CS0 Base Address (BAR2)

Bits	Description
31:3	**SecondaryIDECS0BaseAddress: Secondary IDE CS0 Base Address**. Read-write. Reset: 0. Base address[31:3] for Secondary IDE Bus CS0 in IDE mode. This register is used for native mode only. See 3.26.3.2.1 [IDE Compatibility Mode and Native Mode Registers].
2:1	Reserved.
0	**ResourceTypeIndicator: Resource Type Indicator**. Value: 1. 1=Base address field in this register maps to the IO space.

D11F0x1C Secondary IDE CS1 Base Address (BAR3)

Bits	Description
31:2	**SecondaryIDECS1BaseAddress: Secondary IDE CS1 Base Address**. Read-write. Reset: 0. Base address[31:2] for Secondary IDE Bus CS1 in IDE mode. This register is used for native mode only. See 3.26.3.2.1 [IDE Compatibility Mode and Native Mode Registers].
1	Reserved.
0	**ResourceTypeIndicator: Resource Type Indicator**. Value: 1. 1=Base address field in this register maps to the IO space.

D11F0x20 Bus Master Interface Register Base Address (BAR4)

Bits	Description
31:4	**BusMasterInterfaceRegisterBaseAddress: Bus Master Interface Register Base Address.** Read-write. Reset: 0. Base address[31:4] for Bus Master interface registers. See 3.26.3.2.2 [IDE Bus Master Registers].
3:1	Reserved.
0	**ResourceTypeIndicator: Resource Type Indicator**. Value: 1. 1=Base address field in this register maps to the IO space.

D11F0x24 AHCI Base Address (BAR5)

Bits	Description
31:10	**AHCIBaseAddress: AHCI Base Address[31:10]**. Read-write. Reset: 0. Specifies base address[31:10] of the AHCI control register space. See 3.26.3.3 [SATA Memory Mapped AHCI Registers].
9:1	Reserved.
0	**ResourceTypeIndicator: Resource Type Indicator**. Value: 0. 0=Base address field in this register maps to the memory space.

D11F0x2C Subsystem ID and Subsystem Vendor ID

Bits	Description
31:16	**SubsystemID: Subsystem ID**. Read-only. Reset: 1022h.
15:0	**SubsystemVendorID: Subsystem Vendor ID**. Read-only. Reset: 1022h.

D11F0x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapabilitiesPointer: Capabilities Pointer**. Read-only. Reset: 60h. The first pointer of the Capability block.

D11F0x3C Interrupt Line

Bits	Description
31:24	**MaximumLatency: Maximum Latency**. Value: 0. Specifies the Maximum Latency time required before the SATA controller can start an access as a bus-master.
23:16	**MinimumGrant: Minimum Grant**. Value: 0. Specifies the desired settings for how long of a burst the SATA controller needs. The value specifies a period of time in units of 0.25 microseconds.
15:8	**InterruptPin: Interrupt Pin**. Value: 1.
7:0	**InterruptLine: Interrupt Line**. Read-write. Reset: 0. Identifies which input on the interrupt controller this function's PCI interrupt request pin as specified in InterruptPin is routed to.

D11F0x40 Misc Control

Bits	Description
31:24	Reserved.

23:16	**SataPortDisable[7:0]**. Read-write. Reset: 0. BIOS: See 2.17.3.6.1. This field is bit-significant where bit[0] controls port 0. For each bit, 1=The corresponding port is disabled, link/transport layer clocks are shut down. Bit Definition [0] 1=Port 0 disable [1] 1=Port 1 disable [7:2] Reserved
15:1	Reserved.
0	**SubclassCodeWriteEnable: Subclass Code Write Enable**. Read-write. Reset: 0. BIOS: See 2.17.3.1, 2.17.3.8.2. 1=The following registers/fields are programmable: • D11F0x08[ProgramIF[7:0]] • D11F0x08[SubclassCode] • D11F0x50[MMC]

D11F0x44 Watch Dog Control And Status

Bits	Description
31:24	Reserved.
23:16	**WatchdogCounter: Watchdog Counter**. Read-write. Reset: 80h. BIOS: 20h. Specifies the timeout retry count for PCI downstream retries for SATA ports.
15:2	Reserved.
1	**WatchdogTimeoutStatus: Watchdog Timeout Status**. Read; Write-1-to-clear. Reset: 0. 1=The watchdog counter has expired for PCI downstream transaction, and as a result, the transaction was aborted.
0	**WatchdogEnable: Watchdog Enable**. Read-write. Reset: 0. BIOS: 1. 1=Enable the watchdog counter for all the PCI downstream transactions for SATA ports.

D11F0x50 MSI Control

Bits	Description
31:24	Reserved.
23	**C64: MSI 64-bit Address**. Read-only. Reset: 1. 1=64-bit address is supported.
22:20	**MME: Multiple Message Enable**. Read-write. Reset: 0. Indicates the number of messages the HBA should assert. If the value programmed into this field exceeds the MMC field in this register, the results are undetermined.
19:17	**MMC: Multiple Message Capable**. Read-only. Reset: 2h. BIOS: See 2.17.3.8.2. Indicates the number of messages the HBA wishes to assert.
16	**MSIE: Message Signaled Interrupt Enable**. Read-write. Reset: 0. 1=MSI is enabled and the traditional pins are not used to generate interrupts. 0=MSI operation is disabled and the traditional interrupt pins are used.
15:8	**CapabilityNextPointer: Capability Next Pointer**. Read-only. Reset: D0h. BIOS: 0.
7:0	**CapabilityID: Capability ID**. Read-only. Reset: 5h. Indicates this is MSI capability ID.

D11F0x54 MSI Address

Bits	Description
31:2	**MsiAddress**. Read-write. Reset: 0. Specifies the lower 32 bits of the system specified message address [31:2]. Always doubleword aligned.
1:0	Reserved.

D11F0x58 MSI Upper Address

Bits	Description
31:0	**MsiUpperAddress**. Read-write. Reset: 0. Specifies the upper 32 bits of the system specified message address [63:32].

D11F0x5C MSI Data

Bits	Description
31:16	Reserved.
15:0	**MsiData**. Read-write. Reset: 0. MSI Data.

D11F0x60 Power Management Capability

Bits	Description
31:27	**PSUP: PME Support**. Read-only. Reset: 8. Indicates the states that can generate PME#. Bit Definition [0] 1=PME# can be generated form D0. Not supported; only interrupts are used. [1] 1=PME# can be generated form D1. D1 is not supported. [2] 1=PME# can be generated form D2. D2 is not supported. [3] 1=PME# can be generated from D3hot state. [4] 1=PME# can be generated form D3cold state. D3cold is not supported.
26	**D2S: D2 Support**. Read-only. Reset: 0. 0=D2 state is not supported.
25	**D1S: D1 Support**. Read-only. Reset: 0. 0=D1 state is not supported.
24:22	**AUXC: Aux Current**. Value: 0. Reports the maximum Suspend well current required in D3cold state.
21	**DSI: Device Specific Initialization**. Read-only. Reset: 1. 1=Device-specific initialization is required.
20	Reserved.
19	**PMEC: PME Clock**. Read-only. Reset: 0. 0=PCI clock is not required to generate PME#.
18:16	**Version: Version**. Read-only. Reset: 3h. Indicates support for Revision 1.2 of the PCI Power Management Specification.
15:8	**CapabilityNextPointer: Capability Next Pointer**. Read-only. Reset: 70h. Points to SATA Capability.
7:0	**CapabilityID: Capability ID**. Read-only. Reset: 1. Indicates that this pointer is for PCI power management.

D11F0x64 PCI Power Management Control And Status

Bits	Description
31:16	Reserved.
15	**PMES: PME Status**. Read; Set-by-hardware; Write-1-to-clear. Cold reset: 0. This bit is set independent of PMEE according to PCI IPM. Writing '0' has no effect. 1=PME# is generated by HBA.
14:9	Reserved.
8	**PMEE: PME Enable**. Read-write. Reset: 0. 1=The SATA controller asserts the PME# signal when PMES == 1. 0=PME# assertion is disabled.
7:4	Reserved.
3	**NSR: No soft reset**. Value: 0. If the device transitions from D3hot to D0 due to system or bus segment reset and PME is supported and enabled, it returns to the device D0 state uninitialized with only PME context preserved regardless of this bit. 1=Devices do not perform an internal reset upon transitioning from D3hot to D0 because of PowerState commands; Configuration contexts is still preserved; Upon transitioning from D3hot to D0 Initialized state, no additional operating system intervention is required to preserve configuration context beyond writing the PowerState bits. 0=Devices do perform an internal reset upon transitioning from D3hot to D0 via software control of the PowerState bits; Configuration context is lost when performing the soft reset; Upon transition from the D3hot to D0 state, full re-initialization sequence is needed to return the device to D0 initialized.
2	Reserved.
1:0	**PS: Power State**. Read-write. Reset: 0. This field is used both to determine the current power state of the HBA and to set a new power state. The D1 and D2 states are not supported. When the HBA is in D3hot state, the configuration space is available, but the register memory spaces are not; Additionally, interrupts are blocked. Bits / Definition: 00b D0 state. 01b Reserved. 10b Reserved. 11b D3hot state.

D11F0x70 Serial ATA Capability Register 0

Bits	Description
31:24	Reserved.
23:20	**MajorRevision: Major Revision**. Read-only. Reset: 1. Major revision number of the SATA Capability Pointer implemented.
19:16	**MinorRevision: Minor Revision**. Read-only. Reset: 0. Minor revision number of the SATA Capability Pointer implemented.
15:8	**CapabilityNextPointer: Capability Next Pointer**. Read-only. Reset: 50h. BIOS: See 2.17.3.8.2. Points to MSI capability.
7:0	**CapabilityID: Capability ID**. Read-only. Reset: 12h. Indicate that this pointer is for Serial ATA Capability.

D11F0x74 Serial ATA Capability Register 1

Bits	Description
31:24	Reserved.
23:4	**BAROffset: BAR Offset**. Read-only. Reset: 0. Indicates the offset into the BAR where the index/data pair are located in doubleword granularity.
3:0	**BARLocation: BAR Location**. Read-only. Reset: Fh. Indicates that index/data pair is implemented in doublewords directly following SATA Capability Register 1 in the PCI configuration space.

D11F0x78 IDP Index Register

Bits	Description
31:10	Reserved.
9:2	**IDPIndex: IDP Index**. Read-write. Reset: 0. This register selects the doubleword offset of the memory mapped AHCI register to be accessed.
1:0	Reserved.

D11F0x7C IDP Data Register

Bits	Description
31:0	**IDPData: IDP Data**. Read-write. Reset: 0. This register is a window through which data is read or written to the memory mapped AHCI register pointed to by the D11F0x78 [IDP Index Register]. See 3.26.3.3 [SATA Memory Mapped AHCI Registers]. This is not a physical register, the default value for this field is 0000_0000h, once software programs a valid address in D11F0x78 [IDP Index Register], it switches to the register value which IDPIndex pointing to. The AHCI registers can be accessed both through D11F0x24 [AHCI Base Address (BAR5)] and through this index/data pair.

D11F0x7C_x00 HBA Capabilities (CAP)

Bits	Description
31:0	Alias of SATAx00.

D11F0x7C_x04 Global HBA Control (GHC)

Bits	Description
31:0	Alias of SATAx04.

D11F0x7C_x08 Interrupt Status (IS)

Bits	Description
31:0	Alias of SATAx08.

D11F0x7C_x0C Ports Implemented (PI)

Bits	Description
31:0	Alias of SATAx0C.

D11F0x7C_x10 AHCI Version (VS)

Bits	Description
31:0	Alias of SATAx10.

D11F0x7C_x14 Command Completion Coalescing Control (CCC_CTL)

Bits	Description
31:0	Alias of SATAx14.

D11F0x7C_x18 Command Completion Coalescing Ports (CCC_PORTS)

Bits	Description
31:0	Alias of SATAx18.

D11F0x7C_x1C Enclosure Management Location (EM_LOC)

Bits	Description
31:0	Alias of SATAx1C.

D11F0x7C_x20 Enclosure Management Control (EM_CTL)

Bits	Description
31:0	Alias of SATAx20.

D11F0x7C_x24 HBA Capabilities Extended (CAP2)

Bits	Description
31:0	Alias of SATAx24.

D11F0x7C_x28 BIOS/OS Handoff Control and Status (BOHC)

Bits	Description
31:0	Alias of SATAx28.

D11F0x7C_x1[8,0]0 Port 1,0 Command List Base Address (PxCLB)

Bits	Description
31:0	Alias of SATAx1[8,0]0.

D11F0x7C_x1[8,0]4 Port 1,0 Command List Base Upper Address (PxCLBU)

Bits	Description
31:0	Alias of SATAx1[8,0]4.

D11F0x7C_x1[8,0]8 Port 1,0 FIS Base Address (PxFB)

Bits	Description
31:0	Alias of SATAx1[8,0]8.

D11F0x7C_x1[8,0]C Port 1,0 FIS Base Address Upper (PxFBU)

Bits	Description
31:0	Alias of SATAx1[8,0]C.

D11F0x7C_x1[9,1]0 Port 1,0 Interrupt Status (PxIS)

Bits	Description
31:0	Alias of SATAx1[9,1]0.

D11F0x7C_x1[9,1]4 Port 1,0 Interrupt Enable (PxIE)

Bits	Description
31:0	Alias of SATAx1[9,1]4.

D11F0x7C_x1[9,1]8 Port 1,0 Command and Status (PxCMD)

Bits	Description
31:0	Alias of SATAx1[9,1]8.

D11F0x7C_x1[A,2]0 Port 1,0 Task File Data (PxTFD)

Bits	Description
31:0	Alias of SATAx1[A,2]0.

D11F0x7C_x1[A,2]4 Port 1,0 Signature (PxSIG)

Bits	Description
31:0	Alias of SATAx1[A,2]4.

D11F0x7C_x1[A,2]8 Port 1,0 Serial ATA Status (PxSSTS)

Bits	Description
31:0	Alias of SATAx1[A,2]8.

D11F0x7C_x1[A,2]C Port 1,0 Serial ATA Control (PxSCTL)

Bits	Description
31:0	Alias of SATAx1[A,2]C.

D11F0x7C_x1[B,3]0 Port 1,0 Serial ATA Error (PxSERR)

Bits	Description
31:0	Alias of SATAx1[B,3]0.

D11F0x7C_x1[B,3]4 Port 1,0 Serial ATA Active (PxSACT)

Bits	Description
31:0	Alias of SATAx1[B,3]4.

D11F0x7C_x1[B,3]8 Port 1,0 Command Issue (PxCI)

Bits	Description
31:0	Alias of SATAx1[B,3]8.

D11F0x7C_x1[B,3]C Port 1,0 SNotification (PxSNTF)

Bits	Description
31:0	Alias of SATAx1[B,3]C.

D11F0x7C_x1[C,4]0 Port 1,0 FIS-based Switching Control (PxFBS)

Bits	Description
31:0	Alias of SATAx1[C,4]0.

D11F0x80 PHY Core Control 1

BIOS: See 2.17.3.4.

Bits	Description
31:9	Reserved.
8	**WRALL: Write Settings To All Ports**. Write-only. Reset: 0. Allow software to simultaneously update all ports when writing to the port dependent PHY fine-tune setting register: D11F0x98. 1=Writes to the PHY fine tune setting registers update all ports with the same value at the same time. 0=Writes to the PHY fine tune setting registers only update individual port selected by PN.
7:6	Reserved.
5:4	**GEN: Generation Speed**. Read-write. Reset: 1. Selects the Generation Speed when software reads/writes to the generation speed dependent PHY fine-tune setting register: D11F0x98. Bits / Definition / Bits / Definition 00b / Reserved / 10b / Gen 2 01b / Gen 1 / 11b / Gen 3
3	Reserved.
2:0	**PN: PHY Fine Tune Target Port**. Read-write. Reset: 0. Selects the port when software reads/writes to the port dependent PHY fine-tune setting register: D11F0x98. Bits / Definition 0h / Port 0 is selected 1h / Port 1 is selected 7h-2h / Reserved

D11F0x84 PHY Core Control 2

Bits	Description
31	Reserved.
30	**DShutDwn: PHY PLL Dynamic Shutdown**. Read-write. Reset: 1. Turn on/off the advanced power saving feature. Bits / Definition 0b / No PLL dynamic shutdown; SATA PHY PLL consumes power even though all the ports are inactive. 1b / SATA PHY PLL dynamically shutdown when all the ports are inactive. This inactivity includes Slumber Mode and port disable, but excludes Partial Mode.
29:27	Reserved.
26	**S5ShadowLdDis: S5 Shadow Register Reload Disable**. Read-write. Reset: 1. BIOS: See 2.17.3.7. 1=Not to load S5 shadow registers back to PHY related control registers when S0 domain loses power other than cold boot. 0=Turn on shadow register reloading.
25:3	Reserved.
2	**RSTB: PHY Global Reset**. Read-write. Reset: 1. BIOS: See 2.17.3.5. 0=Reset all port logic in PHY.
1:0	Reserved.

D11F0x88 PHY Global Control 1

Bits	Description
31:28	**PllVcoTune: PLL VCO Range Select Bits**. Read-write. Reset: 0. BIOS: See 2.17.3.3. PLL VCO range select bits.
27:0	Reserved.

D11F0x8C PHY Global Control 2

Bits	Description
31:8	Reserved.
7:0	**PLL_CLKF: PLL Clock Divider Setting**. Read-write. Reset: 7Dh. BIOS: See 2.17.3.5. Feedback clock divider setting.

D11F0x98 PHY Fine Tune PortX GenX Setting

This register is both port and generation speed sensitive. Software must program D11F0x80[PN] to select port and D11F0x80[GEN] to select generation speed before accessing this register.

Bits	Description
31:17	Reserved.
16:12	**TX_SLEW_CNTRL: Transmitter Output Slew Control**. Read-write. Reset: 3h. BIOS: See 2.17.3.4. Rise/Fall time is controlled by the combined force of TX_DRV_STR, TX_SLEW_CNTRL, and TX_DEEMPH_STR. Depending on the channel loss, the value of TX_SLEW_CNTRL and TX_DEEMPH_STR may have to be adjusted based on package and board design.
11:0	Reserved.

D11F0x9C PortX Setting 2

Bits	Description
31:12	Reserved.
11:9	**RX_SQDET_TH: Squelch Detector Threshold Control**. Read-write. Reset: IF (D11F0x80[GEN] == 1) THEN 4h. ELSEIF (D11F0x80[GEN] == 2) THEN 4h. ELSEIF (D11F0x80[GEN] == 3) THEN 4h. ENDIF. BIOS: See 2.17.3.4. Specifies the squelch detector threshold adjustment. Bits / Definition / Bits / Definition 000b 0 100b +25 mV 001b +6 mV 101b +30 mV 010b -16 mV 110b +14 mV 011b -7 mV 111b +20 mV
8:0	Reserved.

D11F0xB4 PortX BIST Control/Status

Software must program D11F0xB7 [PortX BIST Port Select] to select the port before reading from or writing to this register.

Bits	Description
15:14	Reserved.
13:12	**PortXLinkBistSpeed: PortX Link BIST Speed**. Read-write. Reset: 0. PHY PortX speed control for Link BIST mode. Bits / Definition / Bits / Definition 00b Gen 1 — 10b Gen 2 01b Gen 1 — 11b Gen 3
11:6	Reserved.
5:2	**PortXLinkBistPattern: PortX Link BIST Pattern**. Read-write. Reset: 0. Bits — Definition 0000b — Pseudo-random with ALIGN insertion (when Error Count is used, must choose this pattern). 0001b — D10.2 Highest frequency (for Rx eye diagram measurement). 0010b — SYNC primitive (for Rx eye diagram measurement). 0011b — Reserved. 0100b — Mid Frequency Test Pattern (MFTP). 0101b — Reserved. 0110b — Reserved. 0111b — Forced T-mode enable. T-mode is defined as Far end transmit only mode without Device initiating. In T-mode, the BIST pattern generated is based on the values programed in D11F0xBC and D11F0xC0. 1000b — Reserved. 1001b — Reserved. 1010b — Reserved. 1111b-1011b — Reserved.
1	Reserved.
0	**PortXLinkBistEnable: PortX Link BIST Enable**. Read-write. Reset: 0. 1=PortX is in link BIST mode overriding normal operation and device OOB is ignored by host. To resume normal operation, this bit needs to be cleared to 0 and a port reset is required.

D11F0xB7 PortX BIST Port Select

Bits	Description
7:2	Reserved.
1:0	**PortXBistPortSelect: PortX BIST Port Select**. Read-write. Reset: 0. 1=The port that software reads from or writes to during link BIST. For reads, the port is determined by the last significant bit with value 1 in this field. For writes, the port(s) is determined by the any bit with value 1 in this field. This register must be programmed before writing to/reading from D11F0xB4 to avoid reading from/written to an un-expected port. Bit Definition [0] SATA Port 0. [1] SATA Port 1.

D11F0xBC T-Mode BIST Transit Pattern DW1

Bits	Description
31:0	**TmodeBistTransitPatternDW1: T-mode BIST Transit Pattern DW1**. Read-write. Reset: 0. Transit Pattern DW1. See D11F0xB4[PortXLinkBistPattern] for T-mode enable.

D11F0xC0 T-Mode BIST Transit Pattern DW2

Bits	Description
31:0	**TmodeBistTransitPatternDW2: T-mode BIST Transit Pattern DW2**. Read-write. Reset: 0. Transit Pattern DW2. See D11F0xB4[PortXLinkBistPattern] for T-mode enable.

D11F0xC4 T-Mode BIST Transit Control

Bits	Description
31:3	Reserved.
2	**TmodePbit: T-mode P bit**. Read-write. Reset: 0. The transmit primitives bit.
1	**TmodeSbit: T-mode S bit**. Read-write. Reset: 0. Scrambling bypass.
0	**TmodeAbit: T-mode A bit**. Read-write. Reset: 0. ALIGN primitives bypass mode.

D11F0xD0 Advanced Features Capability Register 0

Bits	Description
31:26	Reserved.
25	**FlrCap**. Read-only. Reset: 1. 1=Indicates support for Function Level Reset (FLR).
24	**TpCap**. Read-only. Reset: 1. 1=Indicates support for the Transactions Pending (TP) bit (D11F0xD4[TP]). TP must be supported if FLR is supported.
23:16	**Length**. Read-only. Reset: 6h. Advanced Feature (AF) Structure Length (Bytes).

15:8	**NextPtr: Next pointer**. Read-only. Reset: 0. 0=End of list.
7:0	**CapID: Capability ID**. Read-only. Reset: 13h. Identifies the function being Advanced Feature (AF) capable.

D11F0xD4 Advanced Features Capability Register 1

Bits	Description
31:9	Reserved.
8	**TP: Transactions Pending**. Read-only; Updated-by-hardware. Reset: 0. 1=The function has issued one or more non-posted transactions which have not been completed, including non-posted transactions that a target has terminated with retry. 0=All non-posted transactions have been completed.
7:1	Reserved.
0	**InitiateFlr**. Read-write; RAZ. Reset: 0. 1=Initiate Function Level Reset (FLR).

D11F0xE0 PCI Target Control TimeOut Counter

Bits	Description
15:8	Reserved.
7:0	**Count: PCI Target Control Timeout Count**. Read-write. Reset: 80h. This field specifies the PCI target control timeout count used to clear any stale target commands to the host controller. Granularity is 15.5 us (Timeout Counter = Count * 15.5 us). The counter is disabled if the count is programmed to 0.

3.26.3.2 SATA IO Registers

3.26.3.2.1 IDE Compatibility Mode and Native Mode Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

Compatibility Mode is selected by D11F0x08[ProgramIF[0], ProgramIF[2]] = 0 respectively. In Compatibility Mode the IO address is forced to the legacy compatible address. The same group of registers can be specified as Native Mode by D11F0x08[ProgramIF[0], ProgramIF[2]] = 1, respectively. In Native Mode, the IO address is specified by D11F0x10, D11F0x14, D11F0x18 and D11F0x1C.

Table 272: IDE Compatibility Mode and Native Mode Address Mapping

Compatibility Mode: Interface, IO Address	Native Mode: Interface, BAR+Offset	Read Function	Write Function	Register
Primary, 1F0h	Primary, D11F0x10+0	Data (16 bit)	Data (16 bit)	IDE[0]_x00
Primary, 1F1h	Primary, D11F0x10+1	Error	Feature	IDE[0]_x01
Primary, 1F2h	Primary, D11F0x10+2	Sector Count	Sector Count	IDE[0]_x02
Primary, 1F3h	Primary, D11F0x10+3	Sector Number	Sector Number	IDE[0]_x03
Primary, 1F4h	Primary, D11F0x10+4	Cylinder Low	Cylinder Low	IDE[0]_x04
Primary, 1F5h	Primary, D11F0x10+5	Cylinder High	Cylinder High	IDE[0]_x05
Primary, 1F6h	Primary, D11F0x10+6	Drive/Head	Drive/Head	IDE[0]_x06

Table 272: IDE Compatibility Mode and Native Mode Address Mapping

Compatibility Mode: Interface, IO Address	Native Mode: Interface, BAR+Offset	Read Function	Write Function	Register
Primary, 1F7h	Primary, D11F0x10+7	Status	Command	IDE[0]_x07
Primary, 3F6h	Primary, D11F0x14+2	Alternate Status	Device Control	IDEA[0]_x02
Secondary, 170h	Secondary, D11F0x18+0	Data (16 bit)	Data (16 bit)	IDE[1]_x00
Secondary, 171h	Secondary, D11F0x18+1	Error	Feature	IDE[1]_x01
Secondary, 172h	Secondary, D11F0x18+2	Sector Count	Sector Count	IDE[1]_x02
Secondary, 173h	Secondary, D11F0x18+3	Sector Number	Sector Number	IDE[1]_x03
Secondary, 174h	Secondary, D11F0x18+4	Cylinder Low	Cylinder Low	IDE[1]_x04
Secondary, 175h	Secondary, D11F0x18+5	Cylinder High	Cylinder High	IDE[1]_x05
Secondary, 176h	Secondary, D11F0x18+6	Drive/Head	Drive/Head	IDE[1]_x06
Secondary, 177h	Secondary, D11F0x18+7	Status	Command	IDE[1]_x07
Secondary, 376h	Secondary, D11F0x1C+2	Alternate Status	Device Control	IDEA[1]_x02

IDE[1:0]x00 IDE DATA

Bits	Description
15:0	**Data**. Read-write. Reset: 0. Accesses to this register must be 16-bit. Addressing as an 8-bit register results in IDE[1:0]x00[15:8] aliasing to IDE[1:0]x01.

IDE[1:0]x01 Feature and Error

Bits	Description
7:0	**FeatureError**. Read-write. Reset: 0.

IDE[1:0]x02 Sector Count

Bits	Description
7:0	**SectorCount**. Read-write. Reset: 0.

IDE[1:0]x03 Sector Number

Bits	Description
7:0	**SectorNumber**. Read-write. Reset: 0.

IDE[1:0]x04 Cylinder Low

Bits	Description
7:0	**CylinderLow**. Read-write. Reset: 0.

IDE[1:0]x05 Cylinder High

Bits	Description
7:0	**CylinderHigh**. Read-write. Reset: 0.

IDE[1:0]x06 Drive and Head

Bits	Description
7:0	**DriveHead**. Read-write. Reset: 0.

IDE[1:0]x07 Command and Status

Bits	Description
7:0	**CommandStatus**. Read-write. Reset: 0.

IDEA[1:0]x02 Device Control and Alternate Status

Bits	Description
7:0	**DevCtrlAltStatus**. Read-write. Reset: 0.

3.26.3.2.2 IDE Bus Master Registers

See section 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

IDE_BMx0[8,0] Bus Master IDE Command

Primary and Secondary Bus Master IDE Command registers.

Bits	Description
7:4	Reserved.
3	**BusMasterReadWrite: Bus Master Read/Write**. Read-write. Reset: 0. This bit should not change during Bus Master transfer cycle, even if terminated by Bus Master IDE Stop. Bits Definition 0b Memory to IDE. 1b IDE to memory.
2:1	Reserved.
0	**BusMasterIDEStartStop: Bus Master IDE Start/Stop**. Read-write. Reset: 0. This bit can not be reset by interrupt from IDE device. This must be reset by software (device driver). Bits Definition 0b Stop. 1b Start.

IDE_BMx0[A,2] Bus-master IDE Status

Primary and Secondary Bus Master IDE Status registers.

Bits	Description
7	**SimplexOnly: Simplex Only**. Value: 0.
6	**SlaveDeviceDMACapable: Slave Device DMA Capable**. Read-write. Reset: 0. Device 1 (Slave) DMA capable.
5	**MasterDeviceDMACapable: Master Device DMA Capable**. Read-write. Reset: 0. Device 0 (Master) DMA capable.
4:3	Reserved.
2	**IDEInterrupt: IDE Interrupt**. Read-write. Reset: 0. 1=An IDE device has asserted its interrupt line. IRQ14 is used for the primary channel, and IRQ15 is used for the secondary channel. If the Interrupt Status bit is set to 0, by writing a 1 to this bit while the interrupt line is still at the active level, this bit remains 0 until another assertion edge is detected on the interrupt line.
1	**BusMasterDMAError: Bus Master DMA Error**. Read; Write-1-to-clear. Reset: 0. 1=The IDE host controller encounters a target abort, master abort, or parity error while transferring data on the PCI bus.
0	**BusMasterActive: Bus Master Active**. Read-only; Updated-by-hardware. Reset: 0. Bus Master IDE active. This bit is set to 1 when IDE_BMx0[8,0][0] == 1. This bit is set to 0 when IDE_BMx0[8,0][0] == 0 or the last transfer for a region is performed. Bits Definition 0b Not active. 1b Active.

IDE_BMx0[C,4] Descriptor Table Pointer

Primary and Secondary Bus Master IDE Descriptor Table Pointer register.

Bits	Description
31:2	**DescriptorTableBaseAddress: Descriptor Table Base Address**. Read-write. Reset: 0. These bits correspond to Address [31:2].
1:0	Reserved.

3.26.3.3 SATA Memory Mapped AHCI Registers

These are the AHCI memory mapped registers. The base address is defined by D11F0x24 [AHCI Base Address (BAR5)]. See section 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention.

3.26.3.3.1 Generic Host Control Registers

SATAx00 HBA Capabilities (CAP)

Bits	Description
31	**S64A: Supports 64-bit Addressing**. Read-only. Reset: 1. Indicates whether HBA can access 64-bit data structures. 1=HBA makes the 32-bit upper bits of the port DMA descriptor, the PRD Base, and each PRD entry read/write.
30	**SNCQ: Supports Native Command Queuing**. Read-only. Reset: 1. Indicates whether HBA supports Serial ATA native command queuing. 1=HBA can handle DMA Setup FISes natively, and can handle the auto-activate optimization through that FIS.
29	**SSNTF: Supports SNotification Register**. Read-only. Reset: 1. 1=HBA supports the SATAx1[B,3]C [Port 1,0 SNotification (PxSNTF)] register and its associated functionality.
28	**SMPS: Supports Mechanical Presence Switch**. Read-only. Reset: 1. 1=HBA supports mechanical presence switches on its ports for use in hot plug operations. 0=This function is not supported. This value is loaded by the BIOS prior to OS initialization.
27	**SSS: Supports Staggered Spin-up**. Read-only. Reset: 0. 0=This function is not supported. This value is loaded by the BIOS prior to OS initialization.
26	**SALP: Supports Aggressive Link Power Management**. Read-only. Reset: 1. 1=HBA can support auto-generating link requests to the Partial or Slumber states when there are no commands to process.
25	**SAL: Supports Activity LED**. Read-only. Reset: 1. 1=HBA supports a single activity indication output pin; This pin can be connected to an LED on the platform to indicate device activity on any drive.
24	**SCLO: Supports Command List Override**. Read-only. Reset: 1. 1=HBA supports the SATAx1[9,1]8[CLO] bit and its associated function.
23:20	**ISS: Interface Speed Support**. Read-only. Reset: 3h. Indicates the maximum speed HBA can support on its ports. These encodings match the system software programmable SATAx1[A,2]C[SPD] field. Values are: Bits — Definition 0000b — Reserved 0001b — Gen1 (1.5 Gbps) 0010b — Gen1 and Gen2 (3 Gbps) 0011b — Gen1 and Gen2 and Gen3 (6 Gbps) 1111b-0100b — Reserved
19	**SNZO: Supports Non-Zero DMA Offsets**. Read-only. Reset: 0. 0=Indicates HBA can not support non-zero DMA offsets for DMA Setup FISes.
18	**SAM: Supports AHCI mode only**. Read-only. Reset: 0. The SATA controller may optionally support AHCI access mechanisms only. 0=Indicates that in addition to the native AHCI mechanism (via AHCI BAR5), the SATA controller implements a legacy, task-file based register interface such as SFF-8038i. 1=Indicates the SATA controller does not implement a legacy, task-file based register interface.
17	**SPM: Supports Port Multiplier**. Read-only. Reset: 1. Indicates whether HBA can support a port multiplier. 1=A port multiplier using command-based switching is supported. 0=A port multiplier is not supported, and a port multiplier may not be attached to this HBA.
16	**FBSS: FIS-based Switching Supported**. Read-only. Reset: 1. 1=Indicates HBA supports Port Multiplier FIS-based switching. 0=Indicates HBA does not support FIS-based switching.

15	**PMD: PIO Multiple DRQ Block**. Read-only. Reset: 1. 0=HBA only supports single DRQ block data transfers for the PIO command protocol. 1=HBA supports multiple DRQ block data transfers for the PIO command protocol.
14	**SSC: Slumber State Capable**. Read-only. Reset: 1. Indicates whether HBA can support transitions to the Slumber state. 0=Software must not allow the HBA to initiate transitions to the Slumber state via aggressive link power management nor the SATAx1[9,1]8[ICC] field in each port, and the SATAx1[A,2]C[IPM] field in each port must be programmed to disallow device initiated Slumber requests. 1=HBA and device initiated Slumber requests can be supported.
13	**PSC: Partial State Capable**. Read-only. Reset: 1. Indicates whether HBA can support transitions to the Partial state. 0=Software must not allow HBA to initiate transitions to the Partial state via aggressive link power management nor the SATAx1[9,1]8[ICC] field in each port, and the SATAx1[A,2]C[IPM] field in each port must be programmed to disallow device initiated Partial requests. 1=HBA and device initiated Partial requests can be supported.
12:8	**NCS: Number of Command Slots**. Read-only. Reset: 1Fh. 0's based value indicate the number of command slots per port supported by this HBA. A minimum of 1 and a maximum of 32 slots per port can be supported. The same number of command slots are available on each implemented port.
7	**CCCS: Command Completion Coalescing Supported**. Read-only. Reset: 1. 0=Indicates that HBA does not support command completion coalescing and SATAx14 [Command Completion Coalescing Control (CCC_CTL)] and SATAx18 [Command Completion Coalescing Ports (CCC_PORTS)] are not implemented. 1=Indicates that HBA supports command completion coalescing and HBA has implemented the SATAx14 [Command Completion Coalescing Control (CCC_CTL)] and the SATAx18 [Command Completion Coalescing Ports (CCC_PORTS)] global HBA registers.
6	**EMS: Enclosure Management Supported**. Read-only. Reset: 0. 1=Indicates that HBA supports enclosure management and has implemented the SATAx1C [Enclosure Management Location (EM_LOC)] and SATAx20 [Enclosure Management Control (EM_CTL)] global HBA registers. 0=Indicates that the HBA does not support enclosure management and the SATAx1C [Enclosure Management Location (EM_LOC)] and SATAx20 [Enclosure Management Control (EM_CTL)] global HBA registers are not implemented.
5	**SXS: Supports External SATA**. Read-only. Reset: 0. 1=Indicates that HBA has one or more Serial ATA ports that have a signal-only connector (i.e., power is not part of that connector) that is externally accessible. 0=Indicates that the HBA has no Serial ATA ports that have a signal-only connector externally accessible. If this bit is set to 1, software may refer to the SATAx1[9,1]8[ESP] bit to determine whether a specific port has its signal connector externally accessible as a signal-only connector.
4:0	**NP: Number of Ports**. Read-only. Reset: 3h. 0's based value indicating the maximum number of ports supported by the HBA silicon. A maximum of 32 ports can be supported. A value of 0, indicating one port, is the minimum requirement. Note that the number of ports indicated in this field may be more than the number of ports indicated in the SATAx0C[PI].

SATAx04 Global HBA Control (GHC)

Bits	Description
31	**AE: AHCI Enable**. Read-write. Reset: 1. 1=Indicates that communication to HBA shall be via AHCI mechanisms. 0=Software shall only communicate with HBA using legacy mechanisms; FISes are not posted to memory, and no commands are sent via AHCI mechanisms. This can be used by an HBA that supports both legacy mechanisms and AHCI to know when the HBA is running under an AHCI driver. Software shall set this bit to 1 before accessing other AHCI registers.
30:3	Reserved.

2	**MRSM: MSI Revert to Single Message**. Read-only. Reset: 0. 1=HBA requested more than one MSI vector but has reverted to using the first vector only. 0=HBA has not reverted to single MSI mode (i.e., hardware is already in single MSI mode, software has allocated the number of messages requested, or hardware is sharing interrupt vectors if D11F0x50[MME]<D11F0x50[MMC]). HBA may revert to single MSI mode when the number of vectors allocated by the host is less than the number requested. This bit shall only be set to 1 when the following conditions hold: • D11F0x50[MSIE] == 1 (MSI is enabled) • D11F0x50MMC] > 0 (multiple messages requested) • D11F0x50MME] > 0 (more than one message allocated) • D11F0x50MME] != D11F0x50[MMC] (messages allocated not equal to number requested) When this bit is set to 1, single MSI mode operation is in use and software is responsible for clearing bits in the IS register to clear interrupts. This bit shall be cleared to 0 by hardware when any of the four conditions stated is false. This bit is also cleared to 0 when D11F0x50[MSIE] == 1 and D11F0x50[MME] == 0. In this case, hardware has been programmed to use single MSI mode, and is not "reverting" to that mode.
1	**IE: Interrupt Enable**. Read-write. Reset: 0. This global bit enables interrupts from HBA. 0=All interrupt sources from all ports are disabled. 1=Interrupts are enabled.
0	**HR: HBA Reset**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. When set by software, this bit causes an internal reset of HBA. All state machines that relate to data transfers and queuing shall return to an idle condition, and all ports shall be re-initialized via COMRESET (if staggered spin-up is not supported). If staggered spin-up is supported, then it is the responsibility of software to spin-up each port after the reset has completed. When HBA has performed the reset action, it shall reset this bit to 0.

SATAx08 Interrupt Status (IS)

Bits	Description
31:0	**IPS: Interrupt Pending Status**. Read; Write-1-to-clear. Reset: 0. This register is bit significant where bit[0] corresponds to port 0. For each bit, 1=The corresponding port has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. The IPS[X] bit is only defined for ports that are implemented or for the command completion coalescing interrupt defined by SATAx14[INT]. All other bits are reserved. Bit Definition [0] Port 0 has pending interrupt. [1] Port 1 has pending interrupt. [30]-[2] Port <IPS> has pending interrupt. [31] Port 31 has pending interrupt.

SATAx0C Ports Implemented (PI)

Bits	Description
31:0	**PI[31:0]: Port Implemented**. Read-only. Reset: 3h. 1=The corresponding port is available for software to use. 0=The port is not available for software to use. The maximum number of bits set to 1 shall not exceed SATAx00[NP] + 1, although the number of bits set in this register may be fewer than SATAx00[NP] + 1. At least one bit should be set to 1. This field is loaded by BIOS. Bit Definition [0] Port 0 Implemented [1] Port 1 Implemented [31:2] Reserved

SATAx10 AHCI Version (VS)

Bits	Description
31:16	**MJR: Major Version Number**. Read-only. Reset: 1. Indicates the major version is 1.
15:0	**MNR: Minor Version Number**. Read-only. Reset: 300h. Indicates the minor version is 10.

SATAx14 Command Completion Coalescing Control (CCC_CTL)

Bits	Description
31:16	**TV: Timeout Value**. Read-write. Reset: 1. The timeout value is specified in 1 millisecond intervals. The timer accuracy shall be within 5%. hCccTimer is loaded with this timeout value. hCccTimer is only decremented when commands are outstanding on selected ports. The HBA signals a CCC interrupt when hCccTimer has decremented to 0. hCccTimer is reset to the timeout value on the assertion of each CCC interrupt. A timeout value of 0 is reserved.
15:8	**CC: Command Completions**. Read-write. Reset: 1. Specifies the number of command completions that are necessary to cause a CCC interrupt. The HBA has an internal command completion counter, hCccComplete that is incremented by one each time a selected port has a command completion. When hCccComplete is equal to the command completions value, a CCC interrupt is signaled. The internal command completion counter is reset to 0 on the assertion of each CCC interrupt. 0=Disable CCC interrupts being generated based on the number of commands completed (i.e., CCC interrupts are only generated based on the timer in this case).
7:3	**INT: CCC Interrupt**. Read-only. Reset: 2h. Specifies the interrupt used by the Command Completion Coalescing (CCC) feature. This interrupt must be marked as unused in SATAx0C [Ports Implemented (PI)] by having the corresponding bit being set to 0. Thus, the CCC interrupt corresponds to the interrupt for an un-implemented port on the controller. When a CCC interrupt occurs, the corresponding bit in SATAx08 [Interrupt Status (IS)] shall be asserted to 1. This field also specifies the interrupt vector used for MSI.

2:1	Reserved.
0	**EN: CCC CTL Enable**. Read-write. Reset: 0. 0=The Command Completion Coalescing (CCC) feature is disabled and no CCC interrupts are generated. 1=The CCC feature is enabled and CCC interrupts may be generated based on timeout or command completion conditions. Software shall only change the contents of the TV and CC fields in this register when this bit is cleared to 0. On transition of this bit from 0 to 1, any updated values for the TV and CC fields shall take effect.

SATAx18 Command Completion Coalescing Ports (CCC_PORTS)

Bits	Description
31:0	**PRT: Ports**. Read-write. Reset: 0. 1=The corresponding port is part of the command completion coalescing feature. 0=The port is not part of the command completion coalescing feature. Bits set to 1 in this register must also have the corresponding bit set to 1 in SATAx0C [Ports Implemented (PI)]. An updated value for this field shall take effect within one timer increment (1 millisecond). Bit Definition [0] Port 0 [1] Port 1 [31:2] Reserved

SATAx1C Enclosure Management Location (EM_LOC)

Bits	Description
31:16	**OFST: Offset**. Value: 0000h. Specifies the offset of the message buffer in doublewords from the beginning of BAR5 defined by D11F0x24.
15:0	**SZ: Buffer Size**. Value: 0000h. Specifies the size of the transmit message buffer area in doublewords. If both transmit and receive buffers are supported, then the transmit buffer begins at (SATAx1C[OFST] * 4) and the receive buffer directly follows it. If both transmit and receive buffers are supported, both buffers are of the size indicated in the Buffer Size field.

SATAx20 Enclosure Management Control (EM_CTL)

Bits	Description
31:28	Reserved.
27	**ATTR_PM: Port Multiplier Support**. Read-only. Reset: 0. 1=The HBA supports enclosure management messages for devices attached via a Port Multiplier. 0=The HBA does not. When cleared to 0, software should use the Serial ATA enclosure management bridge that is built into many Port Multipliers for enclosure services with these devices.
26	**ATTR_ALHD: Activity LED Hardware Driven**. Read-only. Reset: 0. 1=The HBA drives the activity LED for the LED message type in hardware and does not utilize software settings for this LED. The HBA does not begin transmitting the hardware based activity signal until after software has written CTL_TM to 1 after a reset condition.
25	**ATTR_XMT: Transmit Only**. Value: 0. 1=The HBA only supports transmit messages and does not support receive messages. 0=The HBA supports transmit and receive messages.

24	**ATTR_SMB: Single Message Buffer**. Value: 0. 1=The HBA has one message buffer that is shared for messages to transmit and messages received; Unsolicited receive messages are not supported and it is software's responsibility to manage access to this buffer. 0=There are separate receive and transmit buffers such that unsolicited messages could be supported.
23:20	Reserved.
19	**SUPP_SGPIO: SGPIO Enclosure Management Messages**. Value: 0. 1=The HBA supports the SGPIO register interface message type.
18	**SUPP_SES2: SES-2 Enclosure Management Messages**. Read-only. Reset: 0. 1=The HBA supports the SES-2 message type.
17	**SUPP_SAFTE: SAF-TE Enclosure Management Messages**. Read-only. Reset: 0. 1=The HBA supports the SAF-TE message type.
16	**SUPP_LED: LED Message Types**. Read-only. Reset: 0. 1=The HBA supports the LED message type.
15:10	Reserved.
9	**CTL_RST: Reset**. Read-only. Reset: 0.
8	**CTL_TM: Transmit Message**. Read-only. Reset: 0.
7:1	Reserved.
0	**STS_MR: Message Received**. Read; Write-1-to-clear. Reset: 0. The HBA sets this bit to a 1 when a message is completely received into the message buffer.

SATAx24 HBA Capabilities Extended (CAP2)

Bits	Description
31:6	Reserved.
5	**DESO: DevSleep Entrance from Slumber Only**. Read-only. Reset: 0. 1=The HBA ignores software directed entrance to DevSleep via SATAx1[9,1]8[ICC] unless [SATAx1[A,2]8IPM] = 6h. 0=The HBA may enter DevSleep from any link state (active, Partial, or Slumber).
4	**SADM: Supports Aggressive Device Sleep Management**. Read-only. Reset: 0. 1=The HBA supports hardware assertion of the DEVSLP signal after the idle timeout expires. 0=This function is not supported and software shall treat the PxDEVSLP.ADSE field as reserved.
3	**SDS: Supports Device Sleep**. Read-only. Reset: 0. 1=The HBA supports the Device Sleep feature. 0=DEVSLP is not supported and software shall not set SATAx1[9,1]8[ICC] to 8.
2:1	Reserved.
0	**BOH: BIOS/OS Handoff**. Read-only. Reset: 0. 1=The HBA supports the BIOS/OS handoff mechanism and the HBA has implemented SATAx28 [BIOS/OS Handoff Control and Status (BOHC)]. 0=The HBA does not support the BIOS/OS handoff mechanism and SATAx28 [BIOS/OS Handoff Control and Status (BOHC)] is not implemented.

SATAx28 BIOS/OS Handoff Control and Status (BOHC)

Bits	Description
31:5	Reserved.

4	**BB: BIOS Busy**. Read-write. Reset: 0. This bit is used by the BIOS to indicate that it is busy cleaning up for ownership change.
3	**OOC: OS Ownership Change**. Read; Write-1-to-clear. Reset: 0. This bit is set to 1 when the OOS bit transitions from 0 to 1.
2	**SOOE: SMI on OS Ownership Change Enable**. Read-write. Reset: 0. 1=Enables an SMI when the OOC bit has been set to 1.
1	**OOS: OS Owned Semaphore**. Read-write. Reset: 0. The system software sets this bit to request ownership of the HBA controller. Ownership is obtained when this bit reads 1 and the BOS bit reads 0.
0	**BOS: BIOS Owned Semaphore**. Read-write. Reset: 0. The BIOS sets this bit to establish ownership of the HBA controller. BIOS clears this bit in response to a request for ownership of the HBA by system software via OOS.

3.26.3.3.2 Port Control Registers

The following registers configure the SATA ports and are implemented per port. Each port shall have the same register mapping. Port 0 starts at 100h. Port 1 starts at 180h. An algorithm to determine the offset is as follows: Port offset = 100h + (Asserted Bit Position in SATAx0C [Ports Implemented (PI)] * 80h).

SATAx1[8,0]0 Port 1,0 Command List Base Address (PxCLB)

Bits	Description
31:10	**CLB[31:10]: Command List Base Address [31:10]**. Read-write. Reset: 0. Specifies the lower 32 bits of the physical base address for the command list for this port. This base address is used by HBA to fetch commands to execute. The structure pointed to by this address range is 1K bytes in length. This address must be 1K-byte aligned.
9:0	Reserved.

SATAx1[8,0]4 Port 1,0 Command List Base Upper Address (PxCLBU)

Bits	Description
31:0	**CLBU: Command List Base Address Upper**. Read-write. Reset: 0. Specifies the upper 32 bits of the physical base address for the command list for this port. If 64-bit addressing is not supported, this register reads 0.

SATAx1[8,0]8 Port 1,0 FIS Base Address (PxFB)

Bits	Description
31:8	**FB[31:8]: FIS Base Address [31:8]**. Read-write. Reset: 0. Specifies the lower 32 bits of the physical base address for the received FISes for this port. The structure pointed to by this address range is 256 bytes in length. This address must be 256-byte aligned.
7:0	Reserved.

SATAx1[8,0]C Port 1,0 FIS Base Address Upper (PxFBU)

Bits	Description
31:0	**FBU: FIS Base Address Upper**. Read-write. Reset: 0. Specifies the upper 32 bits of the physical base address for the received FISes for this port. If 64-bit addressing is not supported, this register reads 0.

SATAx1[9,1]0 Port 1,0 Interrupt Status (PxIS)

Bits	Description
31	**CPDS: Cold Port Detect Status**. Read; Write-1-to-clear. Reset: 0.
30	**TFES: Task File Error Status**. Read; Write-1-to-clear. Reset: 0. 1=The error bit at bit[0] of the status field in the received FIS is set to 1 when the status register is updated by the device.
29	**HBFS: Host Bus Fatal Error Status**. Read; Write-1-to-clear. Reset: 0. 1=The HBA has encountered a host bus error that it cannot recover from, such as a bad software pointer. In PCI, it indicates a target or master abort.
28	**HBDS: Host Bus Data Error Status**. Read; Write-1-to-clear. Reset: 0. 1=The HBA has encountered a data error (uncorrectable ECC/parity) when reading from or writing to system memory.
27	**IFS: Interface Fatal Error Status**. Read; Write-1-to-clear. Reset: 0. 1=The HBA has encountered an error on the Serial ATA interface, which caused the transfer to stop.
26	**INFS: Interface Non-fatal Error Status**. Read; Write-1-to-clear. Reset: 0. 1=The HBA has encountered an error on the Serial ATA interface and was able to continue operation.
25	Reserved.
24	**OFS: Overflow Status**. Read; Write-1-to-clear. Reset: 0. 1=The HBA has received more bytes from a device than what was specified in the PRD table for the command.
23	**IPMS: Incorrect Port Multiplier Status**. Read; Write-1-to-clear. Reset: 0. 1=The HBA has received a FIS from a device whose Port Multiplier field did not match what was expected. The IPMS bit may be set during enumeration of devices on a Port Multiplier due to the normal Port Multiplier enumeration process. It is recommended that IPMS only be used after enumeration is complete on the Port Multiplier.
22	**PRCS: PhyRdy Change Status**. Read-only. Reset: 0. 1=The internal PhyRdy signal has changed state. This bit reflects the state of SATAx1[B,3]0[DIAG]. To clear this bit, software must program SATAx1[B,3]0[DIAG] to 0.
21:8	Reserved.
7	**DMPS: Device Mechanical Presence Status**. Read; Write-1-to-clear. Reset: 0. 1=A mechanical presence switch attached to this port has been opened or closed, which may lead to a change in the connection state of the device. This bit is only valid if ((SATAx00[SMPS] == 1) && (SATAx1[9,1]8[MPSP] == 1)).
6	**PCS: Port Connect Change Status**. Read-only. Reset: 0. 0=No change in Current Connect Status. 1=Change in Current Connect Status. This bit reflects the state of SATAx1[B,3]0[DIAG[10]]. This bit is only cleared when SATAx1[B,3]0[DIAG] is cleared.
5	**DPS: Descriptor Processed**. Read; Write-1-to-clear. Reset: 0. 1=A Physical Region Descriptor (PRD) with the I bit set has transferred all of its data.

4	**UFS: Unknown FIS Interrupt**. Read-only. Reset: 0. 1=An unknown FIS was received with the I bit set and has been copied into system memory. This bit is cleared to 0 by software clearing the SATAx1[B,3]0[DIAG] bit to 0. This bit does not directly reflect the SATAx1[B,3]0[DIAG] bit. SATAx1[B,3]0[DIAG] is set immediately when an unknown FIS is detected, whereas this bit is set when that FIS is posted to memory. Software should wait to act on an unknown FIS until this bit is set to 1, or the two bits may become out-of-sync.
3	**SDBS: Set Device Bits Interrupt**. Read; Write-1-to-clear. Reset: 0. 1=A Set Device Bits FIS has been received with the I bit set and has been copied into system memory.
2	**DSS: DMA Setup FIS Interrupt**. Read-write. Reset: 0. 1=A DMA Setup FIS has been received with the I bit set and has been copied into system memory.
1	**PSS: PIO Setup FIS Interrupt**. Read; Write-1-to-clear. Reset: 0. 1=A PIO Setup FIS has been received with the I bit set, and it has been copied into system memory, and the data related to that FIS has been transferred. This bit shall be set even if the data transfer resulted in an error.
0	**DHRS: Device to Host Register FIS Interrupt**. Read; Write-1-to-clear. Reset: 0. 1=A D2H Register FIS has been received with the I bit set, and has been copied into system memory.

SATAx1[9,1]4 Port 1,0 Interrupt Enable (PxIE)

Bits	Description
31	**CPDE: Cold Presence Detect Enable**. Read-only. Reset: 0.
30	**TFEE: Task File Error Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[TFES] == 1)).
29	**HBFE: Host Bus Fatal Error Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[HBFS] == 1)).
28	**HBDE: Host Bus Data Error Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && SATAx1[9,1]0[HBDS] == 1)).
27	**IFE: Interface Fatal Error Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[IFS] == 1)).
26	**INFE: Interface Non-fatal Error Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[INFS] == 1)).
25	Reserved.
24	**OFE: Overflow Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[OFS] == 1)).
23	**IPME: Incorrect Port Multiplier Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[IPMS] == 1)).
22	**PRCE: PhyRdy Change Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[PRCS] == 1)).
21:8	Reserved.
7	**DMPE: Device Mechanical Presence Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[DMPS] == 1)). For systems that do not support a mechanical presence switch, this bit is read-only and returns 0.
6	**PCE: Port Change Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[PCS] == 1)).

5	**DPE: Descriptor Processed Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[DPS] == 1)).
4	**UFE: Unknown FIS Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[UFS] == 1)).
3	**SDBE: Set Device Bits FIS Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[SDBS] == 1)).
2	**DSE: DMA Setup FIS Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[DSS] == 1)).
1	**PSE: PIO Setup FIS Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[PSS] == 1)).
0	**DHRE: Device to Host Register FIS Interrupt Enable**. Read-write. Reset: 0. 1=The HBA generates an interrupt if ((SATAx04[IE] == 1) && (SATAx1[9,1]0[DHRS] == 1)).

SATAx1[9,1]8 Port 1,0 Command and Status (PxCMD)

Bits	Description
31:28	**ICC: Interface Communication Control**. Read-write. Reset: 0. This field is used to control power management states of the interface. If the Link layer is currently in the L_IDLE state, writing to this field shall cause the HBA to initiate a transition to the interface power management state requested. If the Link layer is not currently in the L_IDLE state, writing to this field shall have no effect. Bits — Definition 0h — No-Op, Idle: When software reads this value, it indicates the HBA is ready to accept a new interface control command, although the transition to the previously selected state may not have occurred yet. 1h — Active: This shall cause the HBA to request a transition of the interface into the active state. 2h — Partial: This shall cause the HBA to request a transition of the interface to the Partial state. The SATA device may reject the request and the interface shall remain in its current state. 5h-3h — Reserved 6h — Slumber: This shall cause the HBA to request a transition of the interface to the Slumber state. The SATA device may reject the request and the interface shall remain in its current state. 7h — Reserved 8h — DevSleep: This shall cause the HBA to assert the DEVSLP signal associated with the port; the HBA shall ignore the Device Sleep Idle Timeout value specified by PxDEVSLP.DITO. Software shall only request DevSleep when the interface is in an idle state (i.e., PxCI is cleared to 0h and PxSACT are cleared to 0h); if CAP2.SDS is cleared to '0' or if the interface is not idle at the time the register is written, then the HBA shall not assert the DEVSLP signal and the interface remains in its current state. If CAPS.SDS is set to '1', CAP2.DESO is set to '1', and PxSSTS.IPM is not set to '6h', then the HBA shall not assert the DEVSLP signal and the interface shall remain in its current state. Additionally, the HBA shall not assert the DEVSLP signal until PHYRDY has been achieved (after a previous de-assertion). Fh-9h — Reserved. When system software writes a non-reserved value other than No-Op (0h), the HBA shall perform the action and update this field back to Idle (0h). If software writes to this field to change the state to a state the link is already in (i.e., interface is in the active state and a request is made to go to the active state), the HBA shall take no action and return this field to Idle. If the interface is in a low power state and software wants to transition to a different low power state, software must first bring the link to active and then initiate the transition to the desired low power state with the exception of DEVSLP. The transition to DevSleep may occur from any other state if SATAx24[DESO] == 0. If SATAx24[DESO] == 1, then DevSleep may only be transitioned to if the link is in Slumber.
27	**ASP: Aggressive Slumber/Partial**. Read-write. Reset: 0. When set to 1, and ALPE == 1, the HBA shall aggressively enter the Slumber state when it clears SATAx1[B,3]8 and SATAx1[B,3]4 is cleared or when it clears the SATAx1[B,3]4 register and SATAx1[B,3]8 is cleared. When cleared, and ALPE == 1, the HBA shall aggressively enter the Partial state when it clears the SATAx1[B,3]8 register and the SATAx1[B,3]4 register is cleared or when it clears the SATAx1[B,3]4 register and SATAx1[B,3]8 is cleared. See SATAx00[SALP].

26	**ALPE: Aggressive Link Power Management Enable**. Read-write. Reset: 0. 1=The HBA shall aggressively enter a lower link power state (Partial or Slumber) based upon the setting of the ASP bit. See SATAx00[SALP].
25	**DLAE: Drive LED on ATAPI Enable**. Read-write. Reset: 0. 1=The HBA shall drive the LED pin active for commands regardless of the state of bit ATAPI. 0=The HBA shall only drive the LED pin active for commands if bit ATAPI == 0.
24	**ATAPI: Device is ATAPI**. Read-write. Reset: 0. 1=The connected device is an ATAPI device. This bit is used by the HBA to control whether or not to generate the desktop LED when commands are active.
23	Reserved.
22	**FBSCP: FIS-based Switching Capable Port**. Read-only. Reset: 1. 1=Indicates that this port supports Port Multiplier FIS-based switching.
21	**ESP: External SATA Port**. Read-only. Reset: 0. 0=Indicates that this port's signal connector is not externally accessible on a signal only connector. ESP is mutually exclusive with the HPCP bit in this register.
20	**CPD: Cold Presence Detection**. Read-only. Reset: 0. 0=The platform does not support cold presence detection on this port.
19	**MPSP: Mechanical Presence Switch Attached to Port**. Read-only. Reset: 0. 0=The platform does not support a mechanical presence switch attached to this port.
18	**HPCP: Hot Plug Capable Port**. Read-only. Reset: 0. 0=Indicates that this port's signal and power connectors are not externally accessible via a joint signal and power connector.
17	**PMA: Port Multiplier Attached**. Read-write. Reset: 0. 1=A Port Multiplier is attached to the HBA for this port. 0=A Port Multiplier is not attached to the HBA for this port. Software is responsible for detecting whether a Port Multiplier is present; hardware does not auto-detect the presence of a Port Multiplier.
16	**CPS: Cold Presence State**. Read-only. Reset: 0. The CPS bit reports whether a device is currently detected on this port via cold presence detection. 1=The HBA detects via cold presence that a device is attached to this port. 0=The HBA detects via cold presence that there is no device attached to this port.
15	**CR: Command List Running**. Read-only. Reset: 0. 1=The command list DMA engine for the port is running.
14	**FR: FIS Receive Running**. Read-only. Reset: 0. 1=The FIS Receive DMA engine for the port is running.
13	**MPSS: Mechanical Presence Switch State**. Read-only. Reset: 1. This bit reports the state of a mechanical presence switch attached to this port. If SATAx00[SMPS] == 1 and the mechanical presence switch is closed then this bit is cleared to 0. If SATAx00[SMPS] == 1 and the mechanical presence switch is open then this bit is set to 1. If SATAx00[SMPS] == 0 then this bit is cleared to 0. Software should only use this bit if ((SATAx00[SMPS] == 1) && (SATAx1[9,1]8[MPSP] == 1)).
12:8	**CCS: Current Command Slot**. Read-only; Updated-by-hardware. Reset: 0. This field is valid if SATAx1[9,1]8[ST] == 1 and shall be set to the command slot value of the command that is currently being issued by the HBA. When SATAx1[9,1]8[ST] transitions from 1 to 0, this field shall be reset to 0. After SATAx1[9,1]8[ST] transitions from 0 to 1, the highest priority slot to issue from next is command slot 0. After the first command has been issued, the highest priority slot to issue from next is CCS+1. For example, after the HBA has issued its first command, if CCS == 0 and SATAx1[B,3]8 == 3, the next command that is issued is from command slot 1.
7:5	Reserved.

4	**FRE: FIS Receive Enable**. Read-write. Reset: 0. 1=The HBA may post the received FISes into the FIS receive area pointed to by SATAx1[8,0]8 [Port 1,0 FIS Base Address (PxFB)] (and SATAx1[8,0]C [Port 1,0 FIS Base Address Upper (PxFBU)] for 64-bit HBAs). 0=Received FISes are not accepted by the HBA, except for the first D2H register FIS after the initialization sequence, and no FISes are posted to the FIS receive area. System software must not set this bit until SATAx1[8,0]8 (and SATAx1[8,0]C for 64-bit) have been programmed with a valid pointer to the FIS receive area. If software wishes to move the base, this bit must first be cleared, and software must wait for the FR bit in this register to be cleared.
3	**CLO: Command List Override**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Setting this bit to 1 causes SATAx1[A,2]0[STS[7], STS[3]] to be cleared to 0. This allows a software reset to be transmitted to the device regardless of whether the BSY and DRQ bits are still set in the SATAx1[A,2]0[STS] register. The HBA sets this bit to 0 when SATAx1[A,2]0[STS[7], STS[3]] have been cleared to 0. This bit shall only be set to 1 immediately prior to setting SATAx1[9,1]8[ST] to 1 from a previous value of 0. Setting this bit to 1 at any other time is not supported and results in indeterminate behavior. Software must wait for CLO to be cleared to 0 before setting SATAx1[9,1]8[ST] to 1.
2	**POD: Power On Device**. Read-only. Reset: 1. 1=HBAs does not support cold presence detect and the HBA sets the state of a pin on the HBA to 1 so that it may be used to provide power to a cold-presence detectable port.
1	**SUD: Spin-Up Device**. Read-only. Reset: 1. 1=HBAs does not support staggered spin-up.
0	**ST: Start**. Read-write. Reset: 0. 1=The HBA may process the command list. 0=The HBA may not process the command list. Whenever this bit is changed from 0 to 1, the HBA starts processing the command list at entry 0. Whenever this bit is changed from a 1 to a 0, SATAx1[B,3]8 is cleared by the HBA upon the HBA putting the controller into an idle state. Software should only set this bit after SATAx1[9,1]8[FRE] has been set to 1.

SATAx1[A,2]0 Port 1,0 Task File Data (PxTFD)

Bits	Description
31:16	Reserved.
15:8	**ERR: Error**. Read-only. Reset: 0. Contains the latest copy of the task file error register.
7:0	**STS: Status**. Read-only; Updated-by-hardware. Reset: 7Fh. Contains the latest copy of the task file status register. Fields of note in this register that affect AHCI hardware operation are: Bit — Definition [0] — ERR. 1=Indicates an error during the transfer. [2:1] — CS. Command specific. [3] — DRQ. 1=Indicates a data transfer is requested. [6:4] — CS. Command specific. [7] — BSY. 1=Indicates the interface is busy.

SATAx1[A,2]4 Port 1,0 Signature (PxSIG)

Bits	Description
31:0	**SIG: Signature**. Read-write. Reset: FFFF_FFFFh. Contains the signature received from a device on the first D2H Register FIS. The bit order is as following: Bit — Definition [31:24] — LBA High Register [23:16] — LBA Mid Register [15:08] — LBA Low Register [07:00] — Sector Count Register The register is updated once after each reset sequence.

SATAx1[A,2]8 Port 1,0 Serial ATA Status (PxSSTS)

Bits	Description
31:12	Reserved.
11:8	**IPM: Interface Power Management**. Read-only; Updated-by-hardware. Reset: 0. Indicates the current interface state. Bits — Definition 0h — Device not present or communication not established. 1h — Interface in Active state. 2h — Interface in Partial power management state. 5h-3h — Reserved. 6h — Interface in Slumber power management state. 7h — Reserved. 8h — Interface in DevSleep power management state. Fh-9h — Reserved.
7:4	**SPD: Current Interface Speed**. Read-only; Updated-by-hardware. Reset: 0. Indicates the negotiated interface communication speed. Bits — Definition 0h — Device not present or communication not established. 1h — Generation 1 communication rate negotiated. 2h — Generation 2 communication rate negotiated. 3h — Generation 3 communication rate negotiated. Fh-4h — Reserved.
3:0	**DET: Device Detection**. Read-only; Updated-by-hardware. Reset: 0. Indicates the interface device detection and PHY state. Bits — Definition 0h — No device detected and PHY communication not established. 1h — Device presence detected but PHY communication not established. 2h — Reserved. 3h — Device presence detected and PHY communication established. 4h — PHY in offline mode as a result of the interface being disabled or running in a BIST loopback mode. Fh-5h — Reserved.

SATAx1[A,2]C Port 1,0 Serial ATA Control (PxSCTL)

Bits	Description
31:20	Reserved.
19:16	**PMP: Port Multiplier Port**. Read-only. Reset: 0. This field is not used by AHCI.
15:12	**SPM: Select Power Management**. Read-only. Reset: 0. This field is not used by AHCI.
11:8	**IPM: Interface Power Management Transitions Allowed**. Read-write. Reset: 0. Indicates which power states the HBA is allowed to transition to. If an interface power management state is disabled, the HBA is not allowed to initiate that state and the HBA must respond with power management denial to any request from the device to enter that state. Bits — Definition 0h — No interface restrictions. 1h — Disable transitions to the Partial state. 2h — Disable transitions to the Slumber state. 3h — Disable transitions to both Partial and Slumber states. 4h — Disable transitions to the DevSleep power management state. 5h — Disable transitions to the Partial and DevSleep power management states. 6h — Disable transitions to the Slumber and DevSleep power management states. 7h — Disable transitions to the Partial, Slumber and DevSleep power management states. Fh-8h — Reserved.
7:4	**SPD: Speed Allowed**. Read-write. Reset: 0. Indicates the highest allowable speed of the interface. Bits — Definition 0h — No speed negotiation restrictions. 1h — Limit speed negotiation to Generation 1 communication rate. 2h — Limit speed negotiation to a rate not greater than Generation 2 communication rate. 3h — Limit speed negotiation to a rate not greater than Generation 3 communication rate. Fh-4h — Reserved.
3:0	**DET: Device Detection Initialization**. Read-write. Reset: 0. Controls the HBA's device detection and interface initialization. Bits — Definition 0h — No device detection or initialization action requested. 1h — Perform interface communication initialization sequence to establish communication. This is functionally equivalent to a hard reset and results in the interface being reset and communications re-initialized. While this field is 1h, COMRESET is transmitted on the interface. Software should leave the DET field set to 1h for a minimum of 1 millisecond to ensure that a COMRESET is sent on the interface. 3h-2h — Reserved. 4h — Disable the Serial ATA interface and put PHY in off-line mode. Fh-5h — Reserved. This field may only be modified when SATAx1[9,1]8[ST] == 0. Changing this field while SATAx1[9,1]8[ST] == 1 results in undefined behavior. When SATAx1[9,1]8[ST] == 1, this field should have a value of 0.

SATAx1[B,3]0 Port 1,0 Serial ATA Error(PxSERR)

Bits	Description

31:16	**DIAG: Diagnostics**. Read; Write-1-to-clear. Reset: 0. Contains diagnostic error information for use by diagnostic software in validating correct operation or isolating failure modes: Bit — Definition [0] PhyRdy Change (N). Indicates that the PhyRdy signal changed state. This bit is reflected in the SATAx1[9,1]0[PRCS] bit. [1] Phy Internal Error (I). Indicates that the PHY detected some internal error. This bit is always 0 in the current implementation. [2] Comm Wake (W). Indicates that a Comm Wake signal was detected by the PHY. [3] 10B to 8B Decode Error (B). Indicates that one or more 10B to 8B decoding errors has occurred. [4] Disparity Error (D). This field is not used by AHCI. This bit is always 0 in the current implementation. [5] CRC Error (C). Indicates that one or more CRC errors occurred with the Link Layer. [6] Handshake Error (H). Indicates that one or more R_ERR handshake responses were received in response to frame transmission. Such errors may be the result of a CRC error detected by the recipient, a disparity or 8b/10b decoding error, or other error condition leading to a negative handshake on a transmitted frame. [7] Link Sequence Error (S). Indicates that one or more Link state machine error conditions was encountered. The Link Layer state machine defines the conditions under which the link layer detects an erroneous transition. This bit is always 0 in the current implementation. [8] Transport state transition error (T). Indicates that an error has occurred in the transition from one state to another within the Transport layer since the last time this bit was cleared. This bit is always 0 in the current implementation. [9] Unknown FIS Type (F). Indicates that one or more FISs were received by the Transport layer with good CRC, but had a type field that was not recognized/known. [10] Exchanged (X). Indicates a COMINIT signal was received. This bit is reflected in SATAx1[9,1]0[PCS]. [15:11] Reserved.

15:0	**Err: Error**. Read; Write-1-to-clear. Reset: 0. Contains error information for use by host software in determining the appropriate response to the error condition. Bit — Definition [0] Recovered Data Integrity Error (I). A data integrity error occurred that was recovered by the interface through a retry operation or other recovery action. This bit is set upon any error when a Data FIS is received, including reception FIFO overflow, CRC error, or 10b/8b decoding error. [1] Recovered Communications Error (M). Communications between the device and host was temporarily lost but was re-established. This can arise from a device temporarily being removed, from a temporary loss of PHY synchronization, or from other causes and may be derived from the PhyNRdy signal between the PHY and Link layers. [7:2] Reserved. [8] Transient Data Integrity Error (T). A data integrity error occurred that was not recovered by the interface. This bit is set upon any error when a Data FIS is received, including reception FIFO overflow, CRC error, or 10b/8b decoding error. [9] Persistent Communication or Data Integrity Error (C). A communication error that was not recovered occurred and is expected to be persistent. Persistent communication errors may arise from faulty interconnect with the device, from a device that has been removed or has failed, or a number of other causes. [10] Protocol Error (P). A violation of the Serial ATA protocol was detected. [11] Internal Error (E). The host bus adapter experienced an internal error that caused the operation to fail and may have put the host bus adapter into an error state. The internal error may include a master or target abort when attempting to access system memory, an elasticity buffer overflow, a primitive mis-alignment, a synchronization FIFO overflow, and other internal error conditions. Typically, when an internal error occurs, a non-fatal or fatal status bit in SATAx1[9,1]0 register is also set to give software guidance on the recovery mechanism required. [15:12] Reserved.

SATAx1[B,3]4 Port 1,0 Serial ATA Active (PxSACT)

Bits	Description
31:0	**DS: Device Status**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. This field is bit significant. Each bit corresponds to the TAG and command slot of a native queued command, where bit 0 corresponds to TAG 0 and command slot 0. This field is set by software prior to issuing a native queued command for a particular command slot. Prior to writing SATAx1[B,3]8 TAG to 1, software sets DS TAG to 1 to indicate that a command with that TAG is outstanding. The device clears bits in this field by sending a Set Device Bits FIS to the host. The HBA clears bits in this field that are set to 1 in the SActive field of the Set Device Bits FIS. The HBA only clears bits that correspond to native queued commands that have completed successfully. Software should only write to this field when SATAx1[9,1]8[ST] == 1. This field is cleared when SATAx1[9,1]8[ST] is written from 1 to 0 by software. This field is not cleared by a COMRESET or a software reset.

SATAx1[B,3]8 Port 1,0 Command Issue (PxCI)

Bits	Description
31:0	**CI: Commands Issued**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. This field is bit significant. Each bit corresponds to a command slot, where bit 0 corresponds to command slot 0. This field is set by software to indicate to the HBA that a command has been built in system memory for a command slot and may be sent to the device. When the HBA receives a FIS which clears the BSY, DRQ, and ERR bits for the command, it clears the corresponding bit in this register for that command slot. Bits in this field shall only be set to 1 by software when SATAx1[9,1]8[ST] is set to 1. This field is also cleared when SATAx1[9,1]8[ST] is written from 1 to 0 by software.

SATAx1[B,3]C Port 1,0 SNotification (PxSNTF)

Bits	Description
31:16	Reserved.
15:0	**PMN: PM Notify**. Read; Write-1-to-clear. Reset: 0. This field is bit significant. Each bit corresponds to a Port Multiplier (PM) port number, where bit[0] corresponds to PM port 0 and bit[15] corresponds to PM port 15. This field indicates whether a particular device with the corresponding PM Port number issued a Set Device Bits FIS to the host with the Notification bit set. Individual bits are cleared by software writing 1 to the corresponding bit positions. This field is reset to default on a HBA Reset, but it is not reset by COMRESET or software reset.

SATAx1[C,4]0 Port 1,0 FIS-based Switching Control (PxFBS)

Bits	Description
31:20	Reserved.
19:16	**DWE: Device With Error**. Read-only; Updated-by-hardware. Reset: 0. Set by hardware to the value of the Port Multiplier port number of the device that experienced a fatal error condition. This field is only valid when SATAx1[C,4]0[SDE] == 1.
15:12	**ADO: Active Device Optimization**. Read-only. Reset: 2h. This register exposes the number of active devices, for which the FIS-based switching implementation has been optimized. When there are more devices active than indicated in this field, throughput of concurrent traffic may degrade. For optimal performance, software should limit the number of active devices based on this value. The minimum value for this field shall be 2h, indicating that at least two devices may be active with high performance maintained.
11:8	**DEV: Device To Issue**. Read-write. Reset: 0. Set by software to the Port Multiplier port value of the next command to issue. This field enables hardware to know the port the command is to be issued to without fetching the command header. Software shall not issue commands to multiple Port Multiplier ports on the same write of SATAx1[B,3]8 register.
7:3	Reserved.
2	**SDE: Single Device Error**. Read-only. Reset: 0. 1=When a fatal error condition has occurred, hardware believes the error is localized to one device such that software's first error recovery step should be to utilize the SATAx1[C,4]0[DEC] functionality. 0=When a fatal error condition has occurred, the error applies to the entire port and to clear the error SATAx1[9,1]8[ST] shall be cleared to 0 by software. This bit is cleared by setting SATAx1[C,4]0[DEC] to 1 or clearing SATAx1[9,1]8[ST] to 0.

1	**DEC: Device Error Clear**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. When set to 1 by software, the HBA shall clear the device-specific error condition and the HBA shall flush any commands outstanding for the device that experienced the error, including clearing SATAx1[B,3]8 and SATAx1[B,3]4 bits for that device to 0. When hardware has completed error recovery actions, hardware shall clear the bit to 0. Software shall only set this bit to 1 if ((SATAx1[C,4]0[EN] == 1) && (SATAx1[C,4]0[SDE] == 1)).
0	**EN: Enable**. Read-write. Reset: 0. 1=A Port Multiplier is attached and the HBA shall use FIS-based switching for its communication. 0=FIS-based switching is not being used. Software shall only change the value of this bit when SATAx1[9,1]8[ST] == 0.

SATAx1[C,4]4 Port 1,0 Device Sleep (PxDEVSLP)

Bits	Description
31:29	Reserved.
28:25	**DM: DITO Multiplier**. Read-only. Reset: 0. 0's based value that specifies the DITO multiplier that the HBA applies to the specified DITO value, effectively extending the range of DITO from 1 ms to 16383 ms. The HBA computes the total idle timeout as a product of DM and DITO (i.e., DITOactual = DITO * DM). Bits Definition Bits Definition 0h multiply 1 Eh-2h multiply <DM+1> 1h multiply 2 Fh multiply 16
24:15	**DITO: Device Sleep Idle Timeout.** IF ((SATAx24[SDS] == 0) \|\| (SATAx24[SADM] == 0)) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. This field specifies the amount of the time (in approximate 1 ms granularity) that the HBA shall wait before driving the DEVSLP signal. Hardware reloads its port specific Device Sleep timer with this value each time the port transitions out of the DEVSLP state. For example: DevSleep transistions to Active or SATAx1[C,4]4[ADSE] transitions from 0 to 1. Software shall only set this value when SATAx1[9,1]8[ST] == 0 and SATAx1[C,4]4[ADSE] == 0.
14:10	**MDAT: Minimum Device Sleep Assertion Time**. IF (SATAx24[SDS] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. This field specifies the minimum amount of time (in 1 ms granularity) that the HBA must assert the DEVSLP signal before it may be de-asserted. The nominal value is 10 ms and the minimum is 1 ms depending on device identification information. Software shall only set this value when SATAx1[9,1]8[ST] == 0, SATAx1[C,4]4[ADSE] == 0 and prior to setting SATAx1[9,1]8[ICC] to 8h.
9:2	**DETO: Device Sleep Exit Timeout**. IF (SATAx24[SDS] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. This field specifies the maximum duration (with approximately 1 ms granularity) from DEVSLP de-assertion until the device is ready to accept OOB. The nominal value is 20 ms while the max value is 255 ms depending on device identification information. Software shall only set this value when SATAx1[9,1]8[ST] == 0, SATAx1[C,4]4[ADSE] == 0 and prior to setting SATAx1[9,1]8[ICC] to 8h.

1	**DSP: Device Sleep Present**. Read-write. Reset: 0. 1=The platform supports Device Sleep on this port. 0=The platform does not support Device Sleep on this port. This bit may only be set to 1 if SATAx24[SDS] == 1. DSP is mutually exclusive with the SATAx1[9,1]8[HPCP] bit and SATAx1[9,1]8[ESP] bit.
0	**ADSE: Aggressive Device Sleep Enable**. IF ((SATAx24[SADM] == 1) && (SATAx24[SDS] == 1)) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. 0=The HBA does not enter DevSleep unless software directed via SATAx1[9,1]8[ICC]. 1=The HBA shall assert the DEVSLP signal after the port has been idle (SATAx1[B,3]8 == 0 and SATAx1[B,3]4 == 0) for the amount of time specified by the SATAx1[C,4]4[DITO]; If (SATAx24[DESO] == 1), the HBA shall assert the DEVSLP signal after the port has been idle for the amount of time specified by SATAx1[C,4]4[DITO] and the interface is in Slumber state (SATAx1[A,2]8[IPM] == 6h). This bit shall only be set to 1 if SATAx1[C,4]4[DSP] == 1.

3.26.3.3.3 Enclosure Buffer Management Registers

The Enclosure Buffer Management registers range from (D11F0x24 [AHCI Base Address (BAR5)] + SATAx1C[OFST]*4 + 00h) to (D11F0x24 [AHCI Base Address (BAR5)] + SATAx1C[OFST]*4 + FFh) within the memory mapped space.

SATA_EMx00 Message Header

Bits	Description
31:28	Reserved.
27:24	**MTYPE: Message Type.** Read-write. Reset: 0.
23:16	**DSIZE: Data Size**. Read-write. Reset: 0.
15:8	**MSIZE: Message Size**. Read-write. Reset: 0.
7:0	Reserved.

SATA_EMx04 Write SGPIO Register Request (I)

Bits	Description
31:24	**REG_INDX: Register Index**. Read-write. Reset: 0. Specifies the index of the first register in the bank to write. Currently supported index range is: REG_TYPE REG_INDEX 00h 0-1 03h 0 04h 0-7 0Ch 0 It is software's responsibility to make sure the index of the first register it attempts to access is valid and supported. Hardware does not provide a mechanism to return a status to software indicating that the current operation fails. An out-of-range index for a specific type of register is treated as don't care and none of the data for such index is written.
23:16	**REG_TYPE: Register Type**. Read-write. Reset: 0. Specifies the bank of registers to write. Currently supported Register Types are: Bits Definition 00h SGPIO Configuration 02h-01h Reserved 03h Transmit 04h General Purpose Transmit 0Bh-05h Reserved 0Ch AMD Configuration FFh-0Dh Reserved It is software's responsibility to make sure the register bank type it attempts to access is valid and supported. Hardware does not provide a mechanism to return a status to software indicating that the current operation fails.
15:8	**FUNC: Function**. Read-write. Reset: 0.
7:0	**FRAME_TYPE: Frame Type**. Read-write. Reset: 0.

SATA_EMx08 Write SGPIO Register Request (II)

Bits	Description
31:8	Reserved.
7:0	**REG_CNT: Register Count**. Read-write. Reset: 0. Specifies the number of registers starting with the specified index to write. Currently supported register count range is: REG_TYPE REG_CNT 00h 1-2 03h 1 04h 1-8 0Ch 1 It is software's responsibility to make sure the register count is valid and supported. Hardware does not provide a mechanism to return a status to software indicating that the current operation fails. An out-of-range register count for a specific type of register is treated as a don't care and none of the data for such register count is written.

SATA_EMx0C Write SGPIO Register Request (III)

Bits	Description
255:0	**DataRegister[7:0]: Data Register [7:0]**. Read-write. Reset: 0. This field contains 256 bits of data registers ranging from (D11F0x24 [AHCI Base Address (BAR5)] + SATAx1C[OFST] * 4 + 0Ch) to (D11F0x24 [AHCI Base Address (BAR5)] + SATAx1C[OFST] * 4 + 2Bh). Once SATA_EMx04[REG_TYPE], SATA_EMx04[REG_INDX], and SATA_EMx08[REG_CNT] are specified, the values in this field is written to the internal copy of SGPIO registers.

3.26.4 USB Controllers

3.26.4.1 USB 2.0 (EHCI)

The EHCI controllers can be disabled by programming PMxEF.

3.26.4.1.1 Devices 12h Function 0 (EHCI) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space]for details about how to access this space.

D12F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: Device ID**. Read-only. Reset: 7908h. This 16-bit field is assigned by the device manufacturer and identifies the type of device.
15:0	**VendorID: Vendor ID**. Read-only. Reset: 1022h.

D12F0x04 Status/Command

Bits	Description
31	**DetectedParityError: Detected Parity Error**. Read; Write-1-to-clear. Reset: 0. 1=The EHCI controller detects a parity error.
30	**SignaledSystemError: Signaled System Error**. Read; Write-1-to-clear. Reset: 0. 1=The EHCI controller asserted SERR#.
29	**ReceivedMasterAbort: Received Master Abort**. Read; Write-1-to-clear. Reset: 0. 1=The EHCI controller received a PCI master abort while acting as a PCI master.
28	**ReceivedTargetAbort: Received Target Abort**. Read; Write-1-to-clear. Reset: 0. 1=The EHCI controller generated PCI cycle (EHCI controller is the PCI master) is aborted by a PCI target.
27	**SignaledTargetAbort: Signaled Target Abort**. Read; Write-1-to-clear. Read. Reset: 0. 1=The EHCI controller signals Target Abort.
26:25	**DEVSELtiming: DEVSEL timing**. Value: 01b. 01b=Medium timing.
24	**MasterDataParityError: Master Data Parity Error**. Read; Write-1-to-clear. Reset: 0. 1=EHCI controller detects PERR# being asserted while acting as PCI master, regardless whether PERR# was driven by EHCI controller or not.
23	**FastBack2BackCapable: Fast Back-to-Back Capable**. Value: 1. 1=Indicates Fast Back-to-Back capable.
22	Reserved.
21	**Capable66MHz: 66 MHz Capable**. Value: 1. 1=Indicates 66 MHz capable.
20	**CapabilitiesList: Capabilities List**. Read-only. Reset: 1. 1=Indicates that the value read at D12F0x34 [Capability Pointer] is a pointer in Configuration Space to a linked list of new capabilities.
19	**InterruptStatus: Interrupt Status**. Read-only. Reset: 0. This bit reflects the state of the interrupt in the device/function.
18:11	Reserved.

10	**InterruptDisable: Interrupt Disable**. Read-write. Reset: 0. 0=Enable the assertion of the device/function's INTx# signal. 1=Disable the assertion of the device/function's INTx# signal.
9	**FastBack2BackEnable: Fast Back-to-Back Enable**. Read-write. Reset: 0. 0=Only fast back-to-back transactions to the same agent are allowed. 1=The master is allowed to generate fast back-to-back transactions to different agents.
8	**SERREnable: SERR# Enable**. Read-write. Reset: 0. 0=Disable the SERR# driver. 1=Enable the SERR# driver.
7	RAZ.
6	**ParityEnable: Parity Enable**. Read-write. Reset: 0. 0=The device sets its DetectedParityError bit when an error is detected, but continues normal operations without asserting PERR#. 1=The device must take its normal action when a parity error is detected.
5	**VgaPaletteRegisterAccesses: VGA palette register accesses**. Value: 0. 0=The device treats palette write accesses the same as other accesses.
4	**MemoryWriteAndInvalidateCommand: Memory Write and Invalidate Command**. Read-write. Reset: 0. 0=Memory Write must be used. 1=Masters may generate the command.
3	**SpecialCycle: Special Cycle**. Value: 0. 0=Indicates no Special Cycle support.
2	**BusMaster: Bus Master**. Read-write. Reset: 0. 0=Disable the device from generating PCI accesses. 1=Allow the device to behave as a bus master.
1	**MemorySpaceAccesses: Memory Space Accesses**. Read-write. Reset: 0. 0=Disable the device response. 1=Allow the device to respond to memory space accesses.
0	**IOSpaceAccesses: IO Space Accesses**. Read-write. Reset: 0. 0=Disable the device response. 1=Allow the device to respond to IO space accesses.

D12F0x08 Revision ID / Class Code

Bits	Description
31:24	**BC: Base Class**. Value: 0Ch. 0Ch=Identifies the device being a Serial Bus Controller.
23:16	**SC: Sub Class**. Value: 3h. Sub Class. 03h=Identifies the device being of Universal Serial Bus.
15:8	**PI: Programming Interface**. Value: 20h. 20h=Identifies the device being an EHCI Host Controller.
7:0	**RevisionID: Revision ID**. Value: 49h.

D12F0x0C Miscellaneous

Bits	Description
31:24	**Bist**. Value: 0. 00h=BIST is not supported.
23:16	**HeaderType: Header Type**. Value: 00h. 00h=Indicates EHCI is single function.
15:8	**LatencyTimer[7:0]: Latency Timer**. Read-only. Reset: 0. This field specifies, in units of PCI bus clocks, the value of the Latency Timer for this PCI bus master.
7:0	**CacheLineSize: Cache Line Size**. Read-write. Reset: 0. Specifies the system cache line size in units of doublewords.

D12F0x10 BAR_EHCI

Bits	Description
31:8	**BA: Base Address[31:8]**. Read-write. Reset: 0. Specifies the MMIO Base address of the EHCI USB controller. See 3.26.4.1.2 [EHCI Memory Mapped IO Registers].
7:4	Reserved.
3	**PM: Prefetch Memory**. Value: 0. 0=Non-prefetchable memory region.
2:1	**Type: base address register type**. Read-only. Reset: 00b. 00b=32-bit base address register.
0	**MemSpace: memory space type**. Read-only. Reset: 0. 0=Memory mapped base address.

D12F0x2C Subsystem ID / Subsystem Vendor ID

Bits	Description
31:16	**SubsystemID**. Read; Write-once. Reset: 7808h.
15:0	**SubsystemVendorID**. Read; Write-once. Reset: 1022h.

D12F0x34 Capability Pointer

Bits	Description
31:8	Reserved.
7:0	**CapabilityPointer: Capability Pointer**. Read-only. IF (D12F0x50[PMEDisable] == 0) THEN Reset: C0h. ELSEIF (D12F0x50[MSIDisable] == 0) THEN Reset: D0h. ELSEIF (D12F0x50[FlrEn] == 1) THEN Reset: F0h. ELSE Reset: 00h. ENDIF. Specifies the address of the first element of capability link.

D12F0x3C Interrupt Line

Bits	Description
31:24	**MaxLat**. Value: 00h.
23:16	**MinGnt**. Value: 00h.
15:8	**InterruptPin: Interrupt Pin**. Read-only. Reset: 01h. 01h=INTA#.
7:0	**InterruptLine: Interrupt Line**. Read-write. Reset: 0. Identifies which input on the interrupt controller the function's PCI interrupt request pin is routed.

D12F0x50 EHCI Misc Control

This register is shared among all the EHCI controllers if (D12F0x64[ShareRegisterEnable] == 1).

Bits	Description
31	**PHYAdvancePowerSavingEnable: PHY Advance Power Saving Enable**. Read-write. Reset: 1. 1=Enable the advance PHY power saving feature to save active power from the USB PHY.
30:29	Reserved.

28	**DisableAsyncQHCacheEnhancement: Disable Async QH Cache Enhancement**. Read-write. Reset: 0. 1=Disable async QH/QTD cache enhancement.
27	**DisablePeriodicListCache: Disable Periodic List Cache**. Read-write. Reset: 0. 1=Disable periodic list cache.
26	**DisableAsyncDataCache: Disable Async Data Cache**. Read-write. Reset: 0. 1=Disable async data cache request.
25	**DisableAsyncQHCache: Disable Async QH Cache**. Read-write. Reset: 0. 1=Disable async QH/QTD cache.
24	**AsyncQHCacheThresholdControl: Async QH Cache Threshold Control**. Read-write. Reset: 0. For OUT, 0=Cache only if packet size is greater than 128 Bytes. 1=Cache only if packet size is greater than 64 Bytes. For IN, 0=Cache only if expected packet size is greater than 192 Bytes. 1=Cache only if expected packet size is greater than 96 Bytes.
23	**AsyncParkDisable: async-park disable**. Read-write. Reset: 0. 1=Disable async-park mode.
22	**ForceNakreLoad: Force nak reload zero**. Read-write. Reset: 0. Force nak counter to zero. 0=Don't reload to zero. 1=Force nak counter to zero.
21	**EnableIPGapExt: Enable InterPacket Gap Extension**. Read-write. Reset: 1. 1=Enable inter-packet gap extension in PIE idle state on starting a new transaction.
20	Reserved.
19	**QualifyCacheHit: Qualify Cache Hit.** Read-write. Reset: 0. BIOS: 1. 1=Qualify cache hit with EHCI_RDWRN command.
18	**Cfg_msi_gcg_en: cfg_msi_gcg_enable**. Read-write. Reset: 0. BIOS: 1. 1=Enable MSI/global clock gating bit.
17	**AsyncParkCacheControl: Async Park Cache Control**. Read-write. Reset: 1. 1=Enable async park cache control.
16	Reserved.
15:12	**AsyncParkOUTControl: Async Park OUT Control**. Read-write. Reset: 1. Async Park Mode Count for OUT Packet. Bits — Definition 0h — Standard count as specified by EHCI1x20[AsyncScheduleParkModeCount]. 1h — 8 packets. Fh-2h — Reserved.
11:8	**AsyncParkINControl: Async Park IN Control**. Read-write. Reset: 1. Async Park Mode Count for IN Packet. Bits — Definition 0h — Standard count as specified by EHCI1x20[AsyncScheduleParkModeCount]. 1h — 8 packets. Fh-2h — Reserved.
7	**EnablePerPortChangeEventsCapability: Enable Per-Port Change Events Capability**. Read-write. Reset: 0. Enable the support of Per-Port Change Event Capability. 1=Set EHCI1x08[PerPort-ChangeEventCapability] to 1. 0=Clear EHCI1x08[PerPortChangeEventCapability] to 0.
6	**MSIDisable: MSI Disable**. Read-write. Reset: 0. BIOS: 1. 1=Disable MSI support.
5	**PMEDisable: PME Disable**. Read-write. Reset: 0. 1=Disable PME support.

4	**EhciPmeLevelSignalByResumeWakeupEnable: EHCI PME level signal by resume-wakeup enable**. Read-write. Reset: 0. BIOS: 1. 1=PME is a level signal instead of pulse when resume wake-up event is detected.
3	Reserved.
2	**FlrEn: FLR Enable**. Read-write. Reset: 0. 1=Enable FLR support.
1	**DisableSMI: disable SMI**. Read-write. Reset: 0. 1=Disable EHCI_SMI sent to USB SMI output (to ACPI). 0=Enable.
0	**D3ColdPMEsupport: D3Cold PME support**. Read-write. Reset: 0. BIOS: 1. 1=Enable EHCI host controller PME support at D3Cold. 0=Disable.

D12F0x54 EHCI Spare 1

This register is shared among all the EHCI controllers if (D12F0x64[ShareRegisterEnable] == 1).

Bits	Description
31	**UsbHoldS0ResumeEn**. Read-write. Reset: 0. BIOS: 1. 0=Disable EHCI hold resume enhancement. 1=Enable EHCI hold resume enhancement.
30:27	Reserved.
26	**EnMaskIntrOfD0: Enable masking interrupt out of D0 state**. Read-write. Reset: 0. 0=Disable masking interrupt for all state, including D3hot. 1=Enable masking interrupt out of D0 state.
25:11	Reserved.
10	**ResumePmeD123En: Resume PME in D123 Enable**. Read-write. Reset: 0. 1=PME Status is set only in non-D0 state when resume occurs.
9:7	**EhciL1EarlyExitTimer: EHCI L1 early exit timer**. Read-write. Reset: 5h. Timer to control EHCI L1 exit. When EhciL1EarlyExitEnable is set to 1, this field selects the amount of time before SOF that the EHCI controller requests UMI to exit from L1. Bits Definition Bits Definition 000b 20 us 100b 60 us 001b 30 us 101b 70 us 010b 40 us 110b 80 us 011b 50 us 111b 90 us
6	**EhciL1EarlyExitMode: Ehci L1 early exit mode**. Read-write. Reset: 0. Mode selection bit for EHCI L1 early exit function. 0=Exit signal active in every uFrame if PDC schedule is enabled. 1=Exit signal active in current uFrame when detecting next uFrame has active PDC work.
5	**EhciL1EarlyExitEnable: EHCI L1 early exit enable**. Read-write. Reset: 0. 0=Disable. 1=Enable.
4	**LSConnectionWakeUpEnable: LS connection wake up enable**. Read-write. Reset: 1. 1=Enable wake up from S states for Low Speed (LS) connection. 0=Disable.
3	**EhciEmptyListMode: EHCI empty list mode**. Read-write. Reset: 1. 1=Enable empty list mode. 0=Disable.
2	**EhciBlockCStateEnable: EHCI block-C-state enable**. Read-write. Reset: 0. 1=Enable EHCI sending block-C-state signal to ACPI.
1	Reserved.
0	**EhciDelayBifL1Enable: EHCI delay BIF L1 enable**. Read-write. Reset: 0. 1=Enable EHCI delaying BIF on entering L1 when there are USB transactions on the controller and when USB delaying BIF L1 is enabled in A-link bridge.

D12F0x60 Serial Bus Release Number / FLADJ

Bits	Description
31:14	Reserved.
13:8	**FLADJ: Frame Length Timing Value**. Read-write. Reset: 20h. Each decimal value change to this register corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF micro-frame length) = 59488 + 16*FLAdj.
7:0	**SBRN: Serial Bus Release Number**. Value: 20h.

D12F0x64 Misc Control 2

Bits	Description
31:29	Reserved.
28:24	**PortDisable**. Read; Write-once; Write-1-only. Reset: 0. For each bit, 1=The corresponding port is disabled. Bit Definition Bit Definition [0] Port 0 [1] Port 1 [4:2] Reserved
23:16	Reserved.
15:14	**Port3ForceResetEnable**. Read-write. Reset: 0. Enables "Force Reset in S-state" feature. See: Port0ForceResetEnable.
13:12	**Port2ForceResetEnable**. Read-write. Reset: 0. Enables "Force Reset in S-state" feature. See: Port0ForceResetEnable.
11:10	**Port1ForceResetEnable**. Read-write. Reset: 0. Enables "Force Reset in S-state" feature. See: Port0ForceResetEnable.
9:8	**Port0ForceResetEnable**. Read-write. Reset: 0. Enables "Force Reset in S-state" feature. Bits Definition 00b Force port reset is disabled. 01b Host controller drives the corresponding port to reset state when system entering S3/S4/S5 states only when port is not connected. 10b Force port reset is disabled. 11b Host controller drives the corresponding port to reset state when system entering S3/S4/S5 states regardless of port connection status.
7:2	Reserved.
1	**ShareRegisterEnable**. Read-write. Reset: 1. 1=Enable register sharing among all EHCI controllers; Software can write to any enabled EHCI controller's share register and the same value is reflected into other EHCI controllers shared register. 0=Disable register sharing; The value written to one controller won't reflect in another equivalent controller. The registers that can be shared are: • D12F0x50 • D12F0x54
0	Reserved.

D12F0x70 Over-Current Control 1

Bits	Description
31:16	Reserved.
15:12	**HSPort3OverCurrentControl: HS Port 3 OverCurrent Control**. Read-only. Reset: Fh. See: HSPort0OverCurrentControl.
11:8	**HSPort2OverCurrentControl: HS Port 2 OverCurrent Control**. Read-only. Reset: Fh. See: HSPort0OverCurrentControl.
7:4	**HSPort1OverCurrentControl: HS Port 1 OverCurrent Control**. Read-only. Reset: Fh. See: HSPort0OverCurrentControl.
3:0	**HSPort0OverCurrentControl: HS Port 0 OverCurrent Control**. Read-only. Reset: Fh. Specifies the OverCurrent pin mapping for port 0. There are 8 OverCurrent pins (USB_OC[7:0]), any value greater than 7 disables the OverCurrent function for port 0. Bits / Definition / Bits / Definition 0000b USB_OC0# 0100b USB_OC4# 0001b USB_OC1# 0101b USB_OC5# 0010b USB_OC2# 0110b USB_OC6# 0011b USB_OC3# 0111b USB_OC7# 1111b-1000b Disable OverCurrent

D12F0x74 Misc Control 2

Bits	Description
31:13	Reserved.
12	**frindex_wr_hsk**. Read-write. Reset: 0. BIOS: 1. 1=Enable programmed FRINDEX. 0=Disable programmed FRINDEX.
11:8	Reserved.
7	**isoch_csplit_mdata_last_enh**. Read-write. Reset: 0. BIOS: 1. 0=Retry immediately if MDATA(last). 1=Won't retry if MDATA(last).
6	**xfer_done_after_eof_upd**. Read-write. Reset: 0. BIOS: 1. 0=update QH when eof and without qTD updated if transfer done is after eof. 1=update QH/qTD after transfer is done.
5	**SptShortPktIntEn**. Read-write. Reset: 1. 1=Enable. 0=Disable.
4	Reserved.
3	**ShortPktIntInSplitEn**. Read-write. Reset: 0. BIOS: 1. 1=Enable. 0=Disable.
2	**AlinkPsEnable**. Read-write. Reset: 0. 1=Enable dynamic A-link gating. 0=AlinkClk is not gated.
1:0	Reserved.

D12F0x78 Target Timeout Control

Bits	Description
31:24	**TimeoutTimer: Timeout Timer**. Read-write. Reset: 80h. Timer to control the purge of the delay queue when the master that has initiated the access does not return to complete the transaction. After the timer expires, the queue is invalidated and the next transaction is serviced.

23:8	Reserved.
7:0	**RetryCounter: Retry Counter**. Read-write. Reset: FFh. Counter to control the purge of the delay queue when the downstream access cycle is not completed within certain time. The transaction is target aborted when the counter expired. 0=The retry counter is disabled.

D12F0x80 Hub Config 0

Bits	Description
31:16	**DeviceID**. Read-write. Reset: 7900h. Assigned by the device manufacturer and identifies the type of device.
15:0	**VendorID**. Read-write. Reset: 0438h.

D12F0x84 Hub Config 1

Bits	Description
31	**SelectiveSuspend**. Read-write. Reset: 1. 1=Enable. 0=Disable. This bit allows another non-SetPortSuspend control transfer immediately to hub ports after it initiates SetPortSuspend to a hub port.
30	Reserved.
29	**LSCLK6ENH**. Read-write. Reset: 1. In LS receiving transfer of normal transfer or LPM transfer, this bit enables enhanced conversion between CLK48 and CLK6. 0=Disable. 1=Enable.
28	**Disable_FORCE_FS_MODE**. Read-write. Reset: 1. 0=Force Fs Mode, related hub downstream ports are forced into full speed mode when a high speed capable device is connected. 1=All hub downstream ports behaves as at USB 2.0 spec defines (not forced into full speed mode) during USB bus enumeration.
27:26	Reserved.
25:24	**HubCfgOCDebounce**. Read-write. Reset: 00b. Adjusts the over-current detection (OC_DET) de-bounce logic. Bits Definition 00b OC_DET has de-bounce timer set to 4~5 ms 01b OC_DET has de-bounce timer set to 170~213 us 10b OC_DET has de-bounce timer set to ~3 us 11b OC_DET has de-bounce timer set close to 20 ns
23	**WkOnConEn**. Read-write. Reset: 1. Controls the hub behavior of wake on connect when the hub is in global suspend state. 0=Hub does not initiate resume on the upstream port, nor assert CLK_REQ signal when it detects a connect event. 1=Enable hub to initiate resume on the upstream port and assert CLK_REQ signal when it detects a connect event and the Remote Wakeup Enable feature is enabled.
22	**WkOnDisConEn**. Read-write. Reset: 1. Controls the hub behavior of wake on disconnect when the hub is in global suspend state. 0=Hub does not initiate resume on the upstream port, nor assert CLK_REQ signal when it detects a disconnect event. 1=Enable hub to initiate resume on the upstream port and assert CLK_REQ signal when it detects a disconnect event and the Remote Wakeup Enable feature is enabled.

21	**WkOnOvrCurEn**. Read-write. Reset: 1. Controls the hub behavior of wake on over current when the hub is in global suspend state. 0=Hub does not initiate resume on the upstream port, nor assert CLK_REQ signal when it detects an over current event. 1=Enable hub to initiate resume on the upstream port and assert CLK_REQ signal when it detects an over current event and the Remote Wakeup Enable feature is enabled.
20	**WkOnDevRsmEn**. Read-write. Reset: 1. Controls the hub behavior of wake on device resume when the HUB is in global suspend state. 0=Hub does not initiate resume on the upstream port, nor assert CLK_REQ signal when it detects resume on downstream port. 1=Enable hub to initiate resume on the upstream port and assert CLK_REQ signal when it detects resume on downstream port.
19	**PdbMode**. Read-write. Reset: 1. Specifies how PORT_PDB_P signal is asserted. 0=PORT_PDB_P is asserted when the downstream port state is Not Configured, Powered-off, Disconnected, Disabled or Suspend; PORT_PDB_P is also asserted when U2HUB_TOP is in Global Suspend. 1=PORT_PDB_P is asserted when the downstream port state is Not Configured, Powered-off, Disconnected, Disabled, Suspend or L1-Suspend; PORT_PDB_P is also asserted when U2HUB_TOP is in Global Suspend.
18	**GSusMaskB**. Read-write. Reset: 1. Masks the CLK_REQ signal deassertion. 0=Masked. 1=Not Masked.
17	**PwrConf**. Read-write. Reset: 1. Indicates the hub power configuration. 0=Bus Powered hub configuration. 1=Self/bus powered hub configuration.
16	**LpmEn**. Read-write. Reset: 0. Specifies whether to support the L1 state.
15:0	**BcdDevice[15:0]**. Read-write. Reset: 0018h. This represents the current revision of the hub device. Bit[7:3] represents major revision, and bit[2:0] represents the minor revision.

D12F0x88 HUB Config 2

Bits	Description
31:24	**BcdUsbLow[7:0]**. Read-write. Reset: 00h.
23:16	**FLAdj[7:0]**. Read-write. Reset: 20h. Specifies Frame Length Adjustment value. Each decimal value change to this field corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF micro-frame length) = 59488 + 16*FLAdj.
15:8	**BESLValue[7:0]**. Read-write. Reset: 00h.
7	**VBusM**. Read-write. Reset: 1. Indicates the VBUS state. 0=VBUS not detected. 1=VBUS detected.
6:4	**BESLValid[2:0]**. Read-write. Reset: 0.
3:0	**TL1HubDrvResume2[3:0]**. Read-write. Reset: 1. Sets the length of resume to be driven for the remote wake of the L1 port device.

D12F0x8C HUB Config 3

The reset value depends on the number of Hub Down Stream Ports (DSP).

Table 273: Reset mapping for D12F0x8C

Register	Reset
D12F0x8C	6400_7EC7h

Bits	Description
31:24	**bHubCtrCurrent[7:0].** Read-write.
23:16	**bMaxPower[7:0]**. Read-write.
15:9	**Removable[7:1]**. Read-write. Specifies the removable state for each port. 0=Non-removable port. 1=Removable port.
8	Reserved.
7	**LPwrSt**. Read-write. Indicates the local power state. 0=Local Power Off. 1=Local Power On.
6:4	**N_Port[2:0]**. Read-write. Indicates the number of available ports for use. Hub has 4 ports; the settable range is 4.
3	**Po2PgSel**. Read-write. 0=0 ms. 1=100 ms.
2	**LedB**. Read-write. Specifies port indicator bit support. 0=LED is supported. 1=LED is unsupported.
1	**BusB**. Read-write. Indicates the current operating mode. 0=Bus powered mode. 1=Self powered mode.
0	**GangB**. Read-write. 0=Ganged power switching mode. 1=Individual power switching mode.

D12F0x9C HUB Status

Bits	Description
31:27	Reserved.
26	**L1SpndO**. Read-only. Reset: 0. 1=The upstream port state machine is in the L1_SPND_O state.
25	**SpndO**. Read-only. Reset: 0. 1=The upstream port state machine is in the SPND_O state.
24	**RmWkUpEn**. Read-only. Reset: 0. Indicates the remote wakeup enable bit state. The original signal is not overridden by LPM transactions. 0=Remote wake up is disabled. 1=Remote wake up is enabled.
23:20	Reserved.
19:16	**PrtGreenLedB[3:0]**. Read-only. Reset: X. Specifies the green LED control for each port. For each bit, 0=LED On. 1=LED Off.
15:12	Reserved.
11:8	**PrtAmberLedB[3:0]**. Read-only. Reset: X. Specifies the amber LED control for each port. For each bit, 0=LED On. 1=LED Off.
7:4	Reserved.
3:0	**PrtPowerB[3:0]**. Read-only. Reset: X. Specifies the power control in each port.

D12F0xA0 USB Legacy Support Extended Capability

Bits	Description
31:25	Reserved.
24	**HcOsOwnedSemaphore**. Read-write. Reset: 0. System software sets this bit to request ownership of the EHCI controller. Ownership is obtained when this bit reads as 1 and the HcBiosOwnedSemaphore reads as 0.
23:17	Reserved.

16	**HcBiosOwnedSemaphore**. Read-write. Reset: 0. The BIOS sets this bit to establish ownership of the EHCI controller. System BIOS sets this bit to a 0 in response to a request for ownership of the EHCI controller by system software.
15:8	**NextEhciExtendedCapabilityPointer**. Read-only. Reset: 00h. 00h=The end of the extended capability list.
7:0	**CapabilityID**. Read-only. Reset: 01h. 01h=Legacy Support.

D12F0xA4 USB Legacy Support Control / Status

Bits	Description
31	**SmiOnBar**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=The Base Address Register (BAR) is written.
30	**SmiOnPciCommand**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=The PCI Command Register is written.
29	**SmiOnOsOwnershipChange**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=D12F0xA0[HcOsOwnedSemaphore] transitions from 1 to 0 or 0 to 1.
28:22	Reserved.
21	**SmiOnAsyncAdvance**. Read-only. Reset: 0. This is the shadow bit of EHCI1x24[InterruptOnAsyncAdvance]. Writing 1 to EHCI1x24[InterruptOnAsyncAdvance] clears this bit to 0.
20	**SmiOnHostSystemError**. Read-only. Reset: 0. This is the shadow bit of EHCI1x24[HostSystemError]. Writing 1 to EHCI1x24[HostSystemError] clears this bit to 0.
19	**SmiOnFrameListRollover**. Read-only. Reset: 0. This is the shadow bit of EHCI1x24[FrameListRollover]. Writing 1 to EHCI1x24[FrameListRollover] clears this bit to 0.
18	**SmiOnPortChangeDetect**. Read-only. Reset: 0. This is the shadow bit of EHCI1x24[PortChangeDetect]. Writing 1 to EHCI1x24[PortChangeDetect] clears this bit to 0.
17	**SmiOnUsbError**. Read-only. Reset: 0. This is the shadow bit of EHCI1x24[UsbErrInt]. Writing 1 to EHCI1x24[UsbErrInt] clears this bit to 0.
16	**SmiOnUsbComplete**. Read-only. Reset: 0. This is the shadow bit of EHCI1x24[UsbInt]. Writing 1 to EHCI1x24[UsbInt] clears this bit to 0.
15	**SmiOnBarEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnBAR == 1.
14	**SmiOnPciCommandEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnPciCommand == 1.
13	**SmiOnOsOwnershipEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnOsOwnershipChange == 1.
12:6	Reserved.
5	**SmiOnAsyncAdvanceEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnAsyncAdvance == 1.
4	**SmiOnHostSystemErrorEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnHostSystemError == 1.
3	**SmiOnFrameListRolloverEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnFrameListRollover == 1.
2	**SmiOnPortChangeEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnPortChangeDetect == 1.

1	**SmiOnUsbErrorEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnUsbError == 1.
0	**UsbSmiEnable**. Read-write. Reset: 0. 1=The host controller issues an SMI if SmiOnUsbComplete == 1.

D12F0xC0 PME Capability

Bits	Description
31:27	**PmeSupport[4:0]**. Read-only. Reset: 0Fh. Indicates the power states in which the function may assert PME#. For each bit, 0=the function is not capable of asserting the PME# signal while in that power state. Bit — Definition [0] — 1=PME# can be asserted from D0 [1] — 1=PME# can be asserted from D1 [2] — 1=PME# can be asserted from D2 [3] — 1=PME# can be asserted from D3hot [4] — 1=PME# can be asserted from D3cold
26	**D2Support**. Read-only. Reset: 1. 1=This function supports the D2 Power Management State.
25	**D1Support**. Read-only. Reset: 1. 1=This function supports the D1 Power Management State.
24:22	**AuxCurrent**. Value: 0. Reports the 3.3V auxiliary current requirements for the PCI function. 0=Self-powered.
21	**DSI: device specific initialization.** Value: 0. Indicates whether special initialization of this function is required before the generic class device driver is able to use it.
20	Reserved.
19	**PmeClock: PME clock**. Value: 0. 0=Indicates that no PCI clock is required for the function to generate PME#.
18:16	**Version**. Read-only. Reset: 2h. 2h=Indicates that this function complies with Revision 1.1 of the PCI power management interface specification.
15:8	**NextItemPointer: next item pointer**. Read-only. Reset: D0h. Specifies the address of the next capability structure.
7:0	**CapId: capability ID**. Value: 01h. Identifies the linked list items as being the PCI Power Management registers.

D12F0xC4 PME Control / Status

Bits	Description
31:16	Reserved.
15	**PmeStatus**. Read; Set-by-hardware; Write-1-to-clear. Cold reset: 0. 1=PME# signal is asserted, independent of the state of the PmeEn bit.
14:9	Reserved.
8	**PmeEn**. Read-write. Cold reset: 0. 1=Enables the function to assert PME#. 0=PME# assertion is disabled.
7:2	Reserved.

1:0	**PowerState**. Read-write. Reset: 0. This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. If software attempts to write an unsupported, optional state to this field, the write operation must be completed normally on the bus; however, the data is discarded and no state change occurs. Bits Definition Bits Definition 00b D0 10b D2 01b D1 11b D3hot

D12F0xD0 MSI Control

Bits	Description
31:24	Reserved.
23	**C64: 64-bit Address Capable**. Read-only. IF (EHCI1x08[AddressingCapability64Bit] == 1) THEN Reset: 1. ELSE Reset: 0. ENDIF. 1=Indicates that the EHCI is capable of generating a 64-bit message address. 0=Indicates the EHCI is not capable of generating a 64-bit address.
22:20	**MSIControl: MSI Control**. Read-write. Reset: 0.
19:17	Reserved.
16	**MSIE: MSI Enable**. Read-write. Reset: 0. 1=MSI is enabled and the traditional pins are not used to generate interrupts. 0=MSI operation is disabled and the traditional interrupt pins are used.
15:8	**NextItemPointer: Next Item Pointer**. Read-only. Reset: E4h. Pointer to next capability structure.
7:0	**CapabilityID**. Read-only. Reset: 5h. Indicates this is USB MSI capability ID.

D12F0xD4 MSI Address

Bits	Description
31:2	**MsiAddress: MSI Address**. Read-write. Reset: 0. Specifies system specified message address[31:2].
1:0	Reserved.

D12F0xD8 MSI Upper Address

Bits	Description
31:0	**MsiUpperAddress: MSI Upper Address**. Read-write. Reset: 0. IF (D12F0xD0[C64] == 1) THEN This field specifies the upper 32 bits of system specific message address[63:32]. ELSE This field specifies the MSI data. ENDIF.

D12F0xDC MSI Data

Bits	Description
31:0	**MsiData: MSI Data**. Read-write. Reset: 0. IF (D12F0xD0[C64] == 1) THEN This field specifies the MSI data. ELSE This field is Reserved. ENDIF.

D12F0xF0 Function Level Reset Capability

IF (D12F0x50[FlrEn] == 0) THEN

Bits	Description
31:0	Reserved.

ELSE

Bits	Description
31:26	Reserved.
25	**FunctionLevelResetCapability: Function Level Reset Capability**. Value: 1. 1=Function level reset support.
24	**TransactionPendingCapability**. Value: 1. 1=Transaction pending feature is supported.
23:16	**Length**. Read-only. Reset: 6h. Advanced Feature (AF) structure length (byte).
15:8	**NextItemPointer: Next Item Pointer**. Value: 0. Pointer to next capability structure. 0=This is the final item on the list.
7:0	**CapId**. Read-only. Reset: 13h. 13h=Identifies that the function is Advanced Feature (AF) capable.

ENDIF.

D12F0xF4 Function Level Reset Control

IF (D12F0x50[FlrEn] == 0) THEN

Bits	Description
31:0	Reserved.

ELSE

Bits	Description
31:9	Reserved.
8	**TransactionPending**. Read-only; Updated-by-hardware. Reset: 0. 1=Indicates that the Function has issued one or more non-posted transactions which have not been completed, including non-posted transactions that a target has terminated with Retry. 0=Indicates that all non-posted transactions have been completed.
7:1	Reserved.
0	**InitiateFLR: Initiate FLR**. RAZ; Write-1-only. Reset: 0. 1=Initiates Function Level Reset (FLR).

ENDIF.

3.26.4.1.2 EHCI Memory Mapped IO Registers

The EHCI MMIO base address is specified by D12F0x10[BA]. For details of ECHI USB controller implementation requirements and behavior consult the *Universal Serial Bus Specification 2.0* and *Enhanced Host Controller Interface Specification for Universal Serial Bus*. See 1.2 [Reference Documents].

EHCI1x00 Capability Length

Bits	Description
7:0	**CapLength**. Value: 20h. This register is used as an offset to add to the register base to find the beginning of the Operational Register Space starting at EHCI1x20.

EHCI1x02 HC Interface Version

Bits	Description
15:0	**HciVersion**. Value: 110h. Specifies a BCD encoding of the version number of interface to which this host controller interface conforms.

EHCI1x04 HC Structural Parameters

Table 274: Reset mapping for EHCI1x04

Register	Reset
EHCI1x04	0020_0004h

Bits	Description
31:24	Reserved.
23:20	**DebugPortNumber**. Read-only. This register identifies which of the host controller ports is the debug port. The value is the port number (one-based) of the debug port. A non-zero value in this field indicates the presence of a debug port. The value in this register must not be greater than EHCI1x04[NPorts].
19:17	Reserved.
16	**PortIndicators**. Read-only. Indicates whether the ports support port indicator control. 1=The port status and control registers include a read/writable field for controlling the state of the port indicator.
15:12	**N_CC: number of companion controllers**. Read-only. Indicates the number of companion controllers associated with this USB 2.0 host controller. A value larger than 0 in this field indicates there are companion USB 1.1 host controller(s). Port-ownership handoffs are supported. High-speed, Full-speed and Low-speed devices are supported on the host controller root ports.
11:8	**N_PCC: NumberOfPortsPerCompanionController**. Read-only. Indicates the number of ports supported per companion host controller. It is used to indicate the port routing configuration to system software.
7	**PortRoutingRules**. Read-only. Specifies how all ports are mapped to companion controllers. 0=The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on.
6:5	Reserved.
4	**PPC: PortPowerControl**. Read-only. Specifies whether the host controller implementation includes port power control. 0=The port does not have port power switches. The value of this field affects the functionality of EHCI1x[70,6C,68,64][PortPower].

3:0	**NPorts**. Read-only. Specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space. Bits Definition 0h undefined Fh-1h <NPorts> Ports

EHCI1x08 HC Capability Parameters

Bits	Description
31:20	Reserved.
19	**Frame32PeriodicListCapability: 32-Frame Periodic List Capability**. Read-only. Reset: 0. 0=Software must treat a Frame List Size value of 11b as reserved.
18	**PerPortChangeEventCapability: Per-Port Change Event Capability**. Value: D12F0x50[EnablePerPortChangeEventsCapability]. 1=Host controller supports per-port change events and the associated fields EHCI1x20[PerPortChangeEventsEnable], EHCI1x24 [PortNChangeDetect] and EHCI1x28[PortChangeIntEn]. 0=Per-port change events are not supported; software should treat those fields as reserved.
17	**LinkPowerManagementCapability: Link Power Management Capability**. Read-only. Reset: 0. 0=Link Power Management L1 State is not supported.
16	**HardwarePrefetchCapability: Hardware Prefetch Capability**. Read-only. Reset: 0. 0=Hardware prefetch capability is not supported.
15:8	**EhciExtendedCapabilitiesPointer**. Read-only. Reset: A0h. Specifies the offset in PCI configuration space of the first EHCI extended capability. See D12F0xA0.
7	**IsochronousSchedulingThreshold3**. Read-only. Reset: 0. The value of this field determines the function of IsochronousSchedulingThreshold[2:0].
6:4	**IsochronousSchedulingThreshold[2:0]**. Read-only. Reset: 111b. This field indicates, relative to the current position of the executing host controller, where software can reliably update the isochronous schedule. IF (IsochronousSchedulingThreshold3 == 0) THEN Field value indicates the number of micro-frames of isochronous data structures (one or more) a host controller can hold before flushing the state. ELSE Host software assumes the host controller may cache an isochronous data structure for an entire frame. ENDIF.
3	Reserved.
2	**AsyncScheduleParkCapability**. Read-only. Reset: 1. 1=The host controller supports the park feature for high-speed queue heads in the Asynchronous Schedule.
1	**ProgrammableFrameListFlag**. Read-only. Reset: 1. 0=EHCI1x20[FrameListSize] is read-only; System software must use a frame list length of 1024 elements with this host controller. 1=EHCI1x20[FrameListSize] is read-writeable; System software can specify and use a smaller frame list via EHCI1x20[FrameListSize].
0	**AddressingCapability64Bit**. Read-only. Reset: 0. Specifies the addressing range capability of this implementation. 0=Data structures using 32-bit address memory pointers. 1=Data structures using 64-bit address memory pointers.

EHCI1x20 USB Command

Bits	Description
31:28	Reserved.
27:24	**HostInitiatedResumeDuration: host-initiated resume duration**. Read-only. Reset: 0. This field is used by system software to specify the minimum amount of time the host controller drives the K-state during a host-initiated resume from a LPM state (e.g., L1), and is conveyed to each LPM-enabled device (via the HIRD bits within an LPM Token's bmAttributes field) upon entry into a low-power state. Bits Definition Fh-0h (<HostInitiatedResumeDuration> *75 us) + 50 us
23:16	**InterruptThresholdControl: Interrupt Threshold Control**. Read-write. Reset: 8h. This field is used by system software to select the maximum rate at which the host controller issues interrupts. The only valid values are defined below. Any other value in this register yields undefined results. Bits Definition Bits Definition 00h Reserved 08h 8 micro-frames (default, 1 ms) 01h 1 micro-frame 10h 16 micro-frames (2 ms) 02h 2 micro-frames 20h 32 micro-frames (4 ms) 04h 4 micro-frames 40h 64 micro-frames (8 ms) Software modifications to this bit while (EHCI1x24[HCHalted] == 0) results in undefined behavior.
15	**PerPortChangeEventsEnable: Per-Port Change Events Enable**. IF (EHCI1x08[PerPortChangeEventCapability] == 1) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. 1=Enable the per-port change event capability as defined by EHCI1x24[PortNChangeDetect] and EHCI1x28[PortNChangeEventEnable].
14	**FullySynchronizedPrefetch: Fully Synchronized Prefetch**. Read-only. Reset: 0.
13	**AsynchronousSchedulePrefetchEnable: Asynchronous Schedule Prefetch Enable**. Read-only. Reset: 0.
12	**PeriodicSchedulePrefetchEnable: Periodic Schedule Prefetch Enable**. Read-only. Reset: 0.
11	**AsyncScheduleParkModeEnable: asynchronous schedule park mode enable**. Read-write. Reset: 1. 1=Park mode is enabled. 0=Park mode is disabled.
10	Reserved.
9:8	**AsyncScheduleParkModeCount: Asynchronous Schedule Park Mode Count**. Read-write. Reset: 11b. This field contains a count of the number of successive transactions the host controller is allowed to execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule. Valid values are 1h to 3h. Bits Definition 00b Reserved. 11b-01b <AsyncScheduleParkModeCount> Successive transactions. Writing 00b to this field while (AsyncScheduleParkModeEnable == 1) results in undefined behavior.
7	**LightHostControllerReset: Light Host Controller Reset**. Read-only. Reset: 0. Not implemented.
6	**InterruptOnAsyncAdvanceDoorbell: Interrupt on Async Advance Doorbell**. Read; Cleared-by-hardware; Write-1-only. Reset: 0. This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Software must write a 1 to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets EHCI1x24[InterruptOnAsyncAdvance] to 1 and clears this bit to 0. Writing 1 to this field while (AsynchronousScheduleEnable == 0) results in undefined behavior.

5	**AsynchronousScheduleEnable: Asynchronous Schedule Enable**. Read-write. Reset: 0. This bit controls whether the host controller skips processing the Asynchronous Schedule. 0=Do not process the Asynchronous Schedule. 1=Use EHCI1x38 to access Asynchronous Schedule.
4	**PeriodicScheduleEnable: Periodic Schedule Enable**. Read-write. Reset: 0. This bit controls whether the host controller skips processing the Periodic Schedule. 0=Do not process the Periodic Schedule. 1=Use EHCI1x34 to access the Periodic Schedule.
3:2	**FrameListSize: frame list size**. IF ((EHCI1x08[ProgrammableFrameListFlag]==1) \|\| (EHCI1x08[Frame32PeriodicListCapability] == 1)) THEN Read-write. ELSE RAZ. ENDIF. Reset: 0. This field specifies the size of the frame list. The size the frame list controls which bits in the Frame Index Register should be used for the Frame List Current index. Bits Definition Bits Definition 00b 1024 frames (4096 bytes) 01b 512 frames (2098 bytes) 10b 256 frames (1024 bytes) 11b Reserved or 32 frames (see below) Software may only program a 32 frame list when EHCI1x08[Frame32PeriodicListCapability] == 1, and a 512 or 256 frame list when EHCI1x08[ProgrammableFrameListFlag] == 1.
1	**HCRESET: Host Controller Reset**. Read; Cleared-by-hardware; Write-1-only. Reset: 0. This control bit is used by software to reset the host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset. Software must reinitialize the host controller in order to return the host controller to an operational state. 1=The Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value; Any transaction currently in progress on USB is immediately terminated; A USB reset is not driven on downstream ports; PCI Configuration registers are not affected by this reset; All operational registers, including port registers and port state machines are set to their initial values; Port ownership reverts to the companion host controller(s). Software should not set this bit to a 1 when EHCI1x24[HCHalted] == 0. Attempting to reset an actively running host controller results in undefined behavior.
0	**RunStop: Run/Stop**. Read-write. Reset: 0. 1=Run; the Host Controller proceeds with execution of the schedule as long as this bit is set to 1. 0=Stop; the Host Controller completes the current and any actively pipelined transactions on the USB and then halts within 16 micro-frames after software clears this bit. Writing 1 to this field while (EHCI1x24[HCHalted] == 0) results in undefined behavior.

EHCI1x24 USB Status

Bits	Description
31:16	**PortNChangeDetect: Port-n Change Detect**. IF (EHCI1x08[PerPortChangeEventCapability] == 1) THEN Read; Set-by-hardware; Write-1-to-clear. ELSE Read-only. ENDIF.Reset: 0. This field should only be used by software when (EHCI1x20[PerPortChangeEventsEnable] == 1). Each bit in this field correspond to a port. For each bit, 1=A port change event was detected for that port; see EHCI1x24[PortChangeDetect] for port change event details. EHCI1x04[NPorts] specifies how many ports are exposed by the host controller and thus how many bits in this field are valid. Bit Definition [0] Port 1 [1] Port 2 [14:2] Port <BIT+1> [15] Port 16

15	**AsynchronousScheduleStatus: Asynchronous Schedule Status**. Read-only. Reset: 0. The bit reports the current real status of the Asynchronous Schedule. 0=Status of the Asynchronous Schedule is disabled. 1=Status of the Asynchronous Schedule is enabled. The host controller is not required to immediately disable or enable the Asynchronous Schedule when software transitions EHCI1x20[AsynchronousScheduleEnable]. When this bit and the EHCI1x20[AsynchronousScheduleEnable] are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).
14	**PeriodicScheduleStatus: Periodic Schedule Status**. Read-only. Reset: 0. The bit reports the current real status of the Periodic Schedule. 0=Periodic Schedule is disabled. 1=Periodic Schedule is enabled. The host controller is not required to immediately disable or enable the Periodic Schedule when software transitions EHCI1x20[PeriodicScheduleEnable]. When this bit and the Periodic Schedule Enable bit are the same value, Periodic Schedule is either enabled (1) or disabled (0).
13	**Reclamation**. Read-only. Reset: 0. Indicates an empty asynchronous schedule.
12	**HCHalted**. Read-only. Reset: 1. 0=EHCI1x20[RunStop] is 1 or the host controller has not yet stopped. 1=The host controller has stopped executing as a result of EHCI1x20[RunStop] being set to 0, either by software or by the host controller hardware due to an internal error.
11:6	Reserved.
5	**InterruptOnAsyncAdvance: Interrupt on Async Advance**. Read-write. Reset: 0. 1=Indicate the assertion of interrupt source caused by EHCI1x20[InterruptOnAsyncAdvanceDoorbell].
4	**HostSystemError: Host System Error**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=A serious error occurs during a host system access involving the host controller module. In a PCI system, conditions that set this bit to 1 include PCI Parity error, PCI Master Abort, and PCI Target Abort. When this error occurs, the host controller clears EHCI1x20[RunStop] to prevent further execution of the scheduled TDs.
3	**FrameListRollover: Frame List Rollover**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=EHCI1x2C[FrameIndex] rolls over from its maximum value to zero. The exact Frame List Index value at which the rollover occurs depends on the frame list size.
2	**PortChangeDetect: Port Change Detect**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=One of the following port change event has occurred: • EHCI1x[70,6C,68,64][PortOwner] changed from 0 to 1; • EHCI1x[70,6C,68,64][ForcePortResume] changed from 0 to 1; • EHCI1x[70,6C,68,64][ConnectStatusChange] is set to 1.
1	**UsbErrInt: USB Error Interrupt**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=The completion of a USB transaction results in an error condition (e.g., error counter underflow). If the TD on which the error interrupt occurred also had its IOC bit set, both this bit and the UsbInt bit are set.
0	**UsbInt: USB Interrupt**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=A USB transaction has completed, which results in the retirement of a Transfer Descriptor that had its IOC bit set; A short packet is detected (actual number of bytes received was less than the expected number of bytes).

EHCI1x28 USB Interrupt Enable

Bits	Description
31:16	**PortNChangeEventEnable: Port-n Change Event Enable**. Read-write. Reset: 0. This field is bit significant. Each bit corresponds to a port. For each bit, 1=The host controller issues an interrupt if the corresponding bit in EHCI1x24[PortNChangeDetect] is set to 1. The interrupt is acknowledged by software clearing EHCI1x24[PortNChangeDetect]. EHCI1x04[NPorts] specifies how many ports are exposed by the host controller and thus how many bits in this field are valid. Bit Definition [0] Port 1 [1] Port 2 [14:2] Port <BIT+1> [15] Port 16
15:6	Reserved.
5	**InterruptOnAsyncAdvanceEn: Interrupt on Async Advance Enable**. Read-write. Reset: 0. 1=The host controller issues an interrupt at the next interrupt threshold if EHCI1x24[InterruptOnAsyncAdvance] == 1. The interrupt is acknowledged by software clearing EHCI1x24[InterruptOnAsyncAdvance].
4	**HostSystemErrorEn: Host System Error Enable**. Read-write. Reset: 0. 1=The host controller issues an interrupt if EHCI1x24[HostSystemError] == 1. The interrupt is acknowledged by software clearing EHCI1x24[HostSystemError].
3	**FrameListRolloverEn: Frame List Rollover Enable**. Read-write. Reset: 0. 1=The host controller issues an interrupt if EHCI1x24[FrameListRollover] == 1. The interrupt is acknowledged by software clearing EHCI1x24[FrameListRollover].
2	**PortChangeIntEn: Port Change Interrupt Enable**. Read-write. Reset: 0. 1=The host controller issues an interrupt if EHCI1x24[PortChangeDetect] == 1. The interrupt is acknowledged by software clearing EHCI1x24[PortChangeDetect].
1	**UsbErrIntEn: USB Error Interrupt Enable**. Read-write. Reset: 0. 1=The host controller issues an interrupt at the next interrupt threshold if EHCI1x24[UsbErrInt] == 1. The interrupt is acknowledged by software clearing EHCI1x24[UsbErrInt].
0	**UsbIntEn: USB Interrupt Enable**. Read-write. Reset: 0. 1=The host controller issues an interrupt at the next interrupt threshold if EHCI1x24[UsbInt] == 1. The interrupt is acknowledged by software clearing EHCI1x24[UsbInt].

EHCI1x2C Frame Index

Bits	Description
31:14	Reserved.
13:0	**FrameIndex: Frame Index**. Read-write; Updated-by-hardware. Reset: 0. This register is used by the host controller to index into the periodic frame list. The register updates every 125 microseconds (once each micro-frame). Bits[N:3] are used to select a particular entry in the Periodic Frame List during periodic schedule execution. The number of bits used for the index depends on EHCI1x20[FrameListSize]. This register must be written as a DWord. Byte writes produce undefined results. This register cannot be written unless EHCI1x24[HCHalted] == 1. Writes to this register also affect the SOF value.

EHCI1x30 Control Data Structure Segment

Bits	Description
31:0	**CtrlDSSegment**. IF (EHCI1x08[AddressingCapability64Bit] == 0) THEN RAZ. ELSE Read-write. ENDIF. Reset: 0. Specifies the most significant address bits[63:32] for all EHCI data structures if EHCI1x08[AddressingCapability64Bit] == 1. This register is concatenated with the link pointer from either EHCI1x34, or EHCI1x38, or any control data structure link field to construct a 64-bit address.

EHCI1x34 Periodic Frame List Base Address

Bits	Description
31:12	**BaseAddress: Base Address**. Read-write. Reset: 0. Specifies the beginning address[31:12] of the Periodic Frame List in the system memory. If (EHCI1x08[AddressingCapability64Bit] == 1), the most significant 32 bits of every control data structure address comes from EHCI1x30.
11:0	Reserved.

EHCI1x38 Current Async List Address

Bits	Description
31:5	**LPL: Link Pointer Low**. Read-write. Reset: 0. Specifies the address[31:5] of the next asynchronous queue head to be executed. If (EHCI1x08[AddressingCapability64Bit] == 1), the most significant 32 bits of every control data structure address comes from EHCI1x30. This field may only reference a Queue Head (QH).
4:0	Reserved.

EHCI1x60 Configure Flag

Bits	Description
31:1	Reserved.
0	**CF: Configure Flag**. Read-only. Reset: 1. This bit controls the default port-routing control logic. Software sets this bit as the last action in its process of configuring the host controller. 0=Port routing control logic default-routes each port to an implementation dependent classic host controller. 1=Port routing control logic default-routes all ports to this host controller.

EHCI1x[70,6C,68,64] Port Status Control [4:1]

Bits	Description
31:25	**DeviceAddress: Device Address**. Read-write. Reset: 0. Specifies the 7-bit USB device address for the device attached to and immediately downstream of the associated root port. 0=Indicates no device is present or support for this feature is not present.

<table>
<tr><td>24:23</td><td>SuspendStatus: Suspend Status. Read-only; Updated-by-hardware. Reset: 0. These two bits are used by software to determine whether the most recent L1 suspend request was successful, specifically:

<table>
<tr><th>Bits</th><th>Definition</th></tr>
<tr><td>00b</td><td>Success: State transition was successful (ACK).</td></tr>
<tr><td>01b</td><td>Not Yet: Device was unable to enter the L1 state at this time (NYET).</td></tr>
<tr><td>10b</td><td>Not Supported: Device does not support the L1 state (STALL).</td></tr>
<tr><td>11b</td><td>Timeout/Error: Device failed to respond or an error occurred.</td></tr>
</table>
This field is updated by hardware immediately following the completion of an L1 transition request (via an LPM Token). To avoid any race conditions with hardware, software should only consume the contents of this field when Suspend == 0 (port no longer in L1).</td></tr>
<tr><td>22</td><td>WakeOnOverCurrentEnable: wake on over-current enable. Read-write. Reset: 0. 1=Enables the port to be sensitive to over-current conditions as wake-up events. This field is 0 if (PortPower == 0).</td></tr>
<tr><td>21</td><td>WakeOnDisconnectEnable: wake on disconnect enable. Read-write. Reset: 0. 1=Enables the port to be sensitive to device disconnects as wake-up events. This field is 0 if (PortPower == 0).</td></tr>
<tr><td>20</td><td>WakeOnConnectEnable: wake on connect enable. Read-write. Reset: 0. 1=Enables the port to be sensitive to device connects as wake-up events. This field is 0 if (PortPower == 0).</td></tr>
<tr><td>19:16</td><td>PortTestControl: Port Test Control. Read-write. Reset: 0. When this field is 0, the port is not operating in a test mode. A non-zero value indicates that it is operating in test mode and the specific test mode is indicated by the specific value.

<table>
<tr><th>Bits</th><th>TestMode</th><th>Bits</th><th>TestMode</th></tr>
<tr><td>0000b</td><td>Test mode not enabled</td><td>0100b</td><td>Test Packet</td></tr>
<tr><td>0001b</td><td>Test J_STATE</td><td>0101b</td><td>Test FORCE_ENABLE</td></tr>
<tr><td>0010b</td><td>Test K_STATE</td><td>1111b-0110b</td><td>Reserved</td></tr>
<tr><td>0011b</td><td>Test SE0_NAK</td><td></td><td></td></tr>
</table></td></tr>
<tr><td>15:14</td><td>PortIndicatorControl: Port Indicator Control. IF (EHCI1x04[PortIndicators] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. This field is 0 if PortPower == 0.

<table>
<tr><th>Bits</th><th>Definition</th><th>Bits</th><th>Definition</th></tr>
<tr><td>00b</td><td>Port indicators are off.</td><td>10b</td><td>Green</td></tr>
<tr><td>01b</td><td>Amber</td><td>11b</td><td>Undefined</td></tr>
</table></td></tr>
<tr><td>13</td><td>PortOwner: Port Owner. Read-only. Reset: 0.</td></tr>
<tr><td>12</td><td>PortPower: Port Power. IF (EHCI1x04[PPC] == 0) THEN Read-only. ELSE Read-write. ENDIF. IF (EHCI1x04[PPC] == 0) THEN Reset: 1. ELSE Reset: X. ENDIF. 0=Power is off; the port is non-functional and does not report attaches, detaches, etc. 1=Power is on. When an over-current condition is detected on a powered port and EHCI1x04[PPC] == 1, the PortPower bit in each affected port may be transitioned by the host controller from 1 to 0 (removing power from the port).</td></tr>
<tr><td>11:10</td><td>LineStatus[1:0]: Line Status. Read-only. Reset: X. These bits reflect the current logical levels of the D+ (LineStatus[1]) and D- (LineStatus[0]) signal lines. These bits are used for detection of low-speed USB devices prior to the port reset and enable sequence. This field is valid only when PortEnable == 0 and CurrentConnectStatus == 1.

<table>
<tr><th>Bits</th><th>Definition</th></tr>
<tr><td>00b</td><td>SE0. Not Low-speed device, perform EHCI reset.</td></tr>
<tr><td>01b</td><td>K-state. Low-speed device, release ownership of port.</td></tr>
<tr><td>10b</td><td>J-state. Not Low-speed device, perform EHCI reset.</td></tr>
<tr><td>11b</td><td>Undefined. Not Low-speed device, perform EHCI reset.</td></tr>
</table>
This value of this field is undefined if (PortPower == 0).</td></tr>
</table>

9	**SuspendUsingL1: Suspend using L1**. Read-write. Reset: 0. 0=The host controller employs the legacy (L2) suspend mechanism. 1=If a non-zero value is specified in DeviceAddress, the host controller generates an LPM Token to enter the L1 state whenever software writes a 1 to the Suspend bit, as well as L1 exit timing during any device- or host-initiated resume. Software should only set this bit when the device attached immediately downstream of this root port supports L1 transitions.
8	**PortReset: Port Reset**. Read-write. Reset: 0. 1=Port is in Reset. 0=Port is not in Reset. Software writes 1 to start the bus reset sequence; It must keep this bit at a 1 long enough to ensure the reset sequence completes. Software writes 0 to terminate the bus reset sequence; This bit does not read as a 0 until after the reset has completed. When software writes this bit to a 1, it must also write a 0 to the PortEnable bit. If the port is in high-speed mode after reset is complete, the host controller automatically enables this port (e.g., set the Port Enable bit to a 1). EHCI1x24[HCHalted] should be a 0 before software attempts to use this bit. The host controller may hold Port Reset asserted to a 1 when the HCHalted bit is a 1. This field is 0 if PortPower == 0.
7	**Suspend**. Read; Cleared-by-hardware; Write-1-only. Reset: 0. 0=IF (EHCI1x[70,6C,68,64][PortEnable] == 1) THEN Port is enabled. ELSE Port is disabled. ENDIF. 1=IF (EHCI1x[70,6C,68,64][PortEnable] == 1) THEN Port is in suspend state; The port can be in either L1 or L2 suspend state depending on SuspendUsingL1. ELSE Port is disabled. ENDIF. When in suspend state, downstream propagation of data is blocked on this port, except for port reset. If this bit is written with a 1 when a transaction is in progress then the blocking does not occur until the end of the current transaction. In the suspend state, the port is sensitive to resume detection. Additional status for L1-based transitions is provided to software via the SuspendStatus field. This field is 0 if PortPower == 0. If system software sets this bit to a 1 when the port is not enabled (i.e., PortEnable == 0) the results are undefined. A write of 0 to this bit is ignored by the host controller. The host controller is unconditionally set this bit to a 0 when: • Software sets the ForcePortResume bit to a 0 from a 1. • Software sets the PortReset bit to a 1 from a 0. • Whenever PortPower is zero.
6	**ForcePortResume: Force Port Resume**. Read-write; Set-by-hardware. Reset: 0. 1=Resume detected/driven on port. 0=No resume (K-state) detected/driven on port. This field is 0 if PortPower == 0. This functionality defined for manipulating this bit depends on the value of the Suspend and SuspendUsingL1 bits. For example, if the port is not suspended (Suspend and Enabled bits are a 1) and software transitions this bit to a 1, then the effects on the bus are undefined. The Host Controller sets this bit to a 1 if a J-to-K transition is detected while the port is in the Suspend state. In this case, EHCI1x24[PortChangeDetect] and/or EHCI1x24[PortNChangeDetect] is also be set to 1. Software sets this bit to a 1 to drive resume signaling. If software sets this bit to a 1, the host controller must not set EHCI1x24[PortChangeDetect] and/or EHCI1x24[PortNChangeDetect]. Writing a 0 (from 1) causes the port to return to high-speed mode. This bit remains a 1 until the port has switched to the high-speed idle. The host controller must complete this transition within 2 milliseconds of software setting this bit to a 0. For legacy (L2) transitions, software must appropriately time the Resume and set this bit to a 0 when the appropriate amount of time has elapsed. Software does not need to time resume signaling for L1 transactions as host controller hardware automatically enforces the necessary timing and clear this bit when the port has fully resumed. Software can influence the amount of time hardware drives resume signaling during L1 exit via EHCI1x20[HostInitiatedResumeDuration].

5	**OverCurrentChange: over-current change**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=There is a change to OverCurrentActive.
4	**OverCurrentActive: over-current active**. Read-only; Updated-by-hardware. Reset: 0. 1=This port currently has an over-current condition. 0=This port does not have an over-current condition.
3	**PortEnableDisableChange: Port Enable/Disable Change**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Port enabled/disabled status has changed. 0=No change. For the root hub, this bit gets set to a 1 only when a port is disabled due to the appropriate conditions existing at the EOF2. This field is 0 if PortPower == 0.
2	**PortEnable: port enable**. Read-write; Set-by-hardware. Reset: 0. 1=Enable. 0=Disable. Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a 1 to this field. The host controller only sets this bit to a 1 when the reset sequence determines that the attached device is a high-speed device. Ports can be disabled by either a fault condition (disconnect event or other fault condition) or by host software. The bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events. When the port is disabled, downstream propagation of data is blocked on this port, except for reset. This field is 0 if PortPower == 0.
1	**ConnectStatusChange: Connect Status Change**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. Indicates a change has occurred in the port's CurrentConnectStatus. The host controller sets this bit for all changes to the port device connect status, even if system software has not cleared an existing connect status change. This field is 0 if PortPower == 0. 1=Change in Current Connect Status. 0=No change.
0	**CurrentConnectStatus: Current Connect Status**. Read-only. Reset: 0. This value reflects the current state of the port, and may not correspond directly to the event that caused the ConnectStatusChange to be set to 1. This field is 0 if PortPower == 0. 1=Device is present on port. 0=No device is present.

EHCI1xA4 Packet Buffer Threshold Values

BIOS: 0040_0040h.

Bits	Description
31:24	Reserved.
23:16	**OUTThreshold: OUT Threshold**. Read-write. Reset: 40h. The transmit packet starts at UTMI interface when threshold of internal FIFO for transmit packet is reached. The value represents multiple of 8 bytes. For example, 10h means 128 bytes. The smallest acceptable value is 08h (64 bytes).
15:8	Reserved.
7:0	**INThreshold: IN Threshold**. Read-write. Reset: 40h. The PCI transaction starts when threshold of internal FIFO for receive packet is reached. The value represents multiple of 8 bytes. For example, 10h means 128 bytes. The smallest acceptable value is 08h (64 bytes).

EHCI1xB4 UTMI Control

BIOS: See 2.17.2.4 & 2.17.2.6.

Bits	Description
31:18	Reserved.

17	**VBusy**. Read-only. Reset: 0. 1=Block software writes to bits[16:8] when port router is updating the field.
16:13	**PortNumber: Port Number**. Read-write. Reset: 0. Select the corresponding port PHY or common block to load the VControl bits. Bits Definition Bits Definition 0000b Port 0 0011b Port 3 0001b Port 1 1110b-0100b Reserved 0010b Port 2 1111b Common block
12	**VLoadB**. Read-write. Reset: 1. VControl value load, active low. 0=Load the new VControl value to the PHY or common block specified by PortNumber. VLoadB must be 1 while VControlModeSel is changed.
11	Reserved.
10:7	**VControlModeSel**. Read-write. Reset: 0. Selects the PHY control mode group. There are 8 groups defined for PHY control mode. Bits Definition Bits Definition 0000b Group 0 0110b Group 6. Analog Control 0001b Group 1 0111b Group 7 0010b Group 2 1000b Group 8 0011b Group 3 1001b Group 9 0100b Group 4. CDR DFT control 1111b-1010b Reserved 0101b Group 5. CDR DFT control
6:0	Reserved.

EHCI1xB8 Loopback Test

Bits	Description
31:27	Reserved.
26:21	**StatusOfPortsforPowerUpStateCheck[5:0]: Status of ports for PowerUp State Check.** Read-only. Reset: 0. Status bit to indicate the Power Up state auto-checking result from the individual port. Each bit maps to a corresponding port, from port 0 to port 5. Bit Definition Bit Definition [0] Port 0 [3] Port 3 [1] Port 1 [4] Port 4 [2] Port 2 [5] Port 5 For each bit, 0=PHY Power Up State checking is fail. 1=PHY Power Up State checking is good.
20	**EnablePHYPowerUpStateChecking: Enable PHY PowerUp State Checking**. Read-write. Reset: 0. 1=Enable auto-checking on PHY Power Up State. There is built-in logic to check the PHY default Power Up state to detect manufacturing defects in the PHY macro.
19:12	**GoodReceivedPacketCount: Good Received Packet Count**. Read-only; Updated-by-hardware. Reset: 0. Specified the number of good packets that the host controller received during the loopback test mode. These bits are cleared by clearing EnableLoopBackTest.
11	Reserved.
10	**LoopbackTestDone: Loopback Test Done**. Read-only. Reset: 0. 1=Loopback test is done.
9	**LoopbackTestStatus: Loopback Test Status**. Read-only. Reset: 0. 0=CRC error on loopback receiving data. 1=Good CRC on loopback receiving data.

8	**EnableLoopBackTest: Enable Loop Back Test**. Read-write. Reset: 0. 1=Enable external USB port loopback test. The loopback test is to set one port to TX mode (Test Packet mode) and one port to RX mode (Test SE0_NAK). See EHCI1x[70,6C,68,64][PortTestControl] for information on the tests.
7:4	Reserved.
3:0	**ReceivedPacketCount: Received Packet Count**. Read-write. Reset: 0. RX data packet count. This counter defines the number (in power of 16) of RX data packet that should be checked for the loop back test.

EHCI1xBC EOR MISC Control

Bits	Description
31:24	**ForceTxData[7:0]: Force TX Data [7:0]**. Read-write. Reset: 0. Used to force TxData[7:0] when the port is in TEST_K mode. This can be used to force PHY to generate a desired output pattern for PHY debugging and characterization purposes.
23:15	Reserved.
14	**EHCIDeepBlinkPowerSavingEnable: EHCI Deep Blink Power Saving Enable**. Read-write. Reset: 1. 1=Enable Deep Blink power saving clock gating; EHCI requests to stop the Global Blink clock when the list processor is idle.
13	Reserved.
12	**EHCIPowerSavingEnable: EHCI Power Saving Enable**. Read-write. Reset: 1. 1=Enable power saving clock gating. When enabled, dynamic clock gating is enabled when EHCI is not in operational mode. The clock going to all memory modules are gated off. Blink clock also gets gated off unless the connection interrupt is detected.
11	**DisablesHSuFrameBabbledetection: Disables HS uFrame Babble detection**. Read-write. Reset: 0. 1=Disable HS uFrame babble detection.
10:8	Reserved.
7:4	**InterPacketGapAdjustCounter: Inter-packet Gap Adjust Counter**. Read-write. Reset: 4h. Specifies the counter used to adjust the inter-packet gap for test packet.
3:2	Reserved.
1	**InterruptrouteControlonForcePortResume: Interrupt route Control on ForcePortResume**. Read-write. Reset: 0. This bit only takes effect when the EnableInterruptOnForcePortResume = 1. 0=Report interrupt to EHCI1x24[UsbInt] bit on software clear EHCI1x[70,6C,68,64][ForcePortResume]. 1=Report interrupt to EHCI1x24[PortChangeDetect] bit on software clear EHCI1x[70,6C,68,64][ForcePortResume].
0	**EnableInterruptonForcePortResume: Enable Interrupt on ForcePortResume**. Read-write. Reset: 0. 1=Enable host controller to generate interrupt on software clear EHCI1x[70,6C,68,64][ForcePortResume].

EHCI1xC0 USB Common PHY CAL and Control

Bits	Description
31	Reserved.
30:24	**CommonPhyCalBus**. Read-only; Updated-by-hardware. Reset: X. PHY Common Calibration Bus.
23	Reserved.

22:20	**ComCalAmp**. Read-only; Updated-by-hardware. Reset: X. Specifies the amplitude calibration data from PHY.
19:18	Reserved.
17	**AddToCommonCalibration**. Read-write. Reset: 1. BIOS: See 2.17.2.5. 1=The signed NewCalBus is added to the ComCalBus and returned to the phy ports; Any overflow is clamped to all 1s; Any underflow is clamped to all 0s. 0=The signed NewCalBus replaces the ComCalBus and returns to the phy ports if UseCommonCalibration = 0.
16	**UseCommonCalibration**. Read-write. Reset: 0. BIOS: See 2.17.2.5. 1=The PHY's calibration value in ComCalBus is returned to the phy ports. 0=The value after adjustment is returned to the PHY ports.
15:8	**NewCalBus[7:0]**. Read-write. Reset: 0. BIOS: See 2.17.2.5. New calibration bus signed value.
7	Reserved.
6:0	**ComCalBus**. Read-only; Updated-by-hardware. Reset: X. Calibration bus value from phy before adjustment.

EHCI1xC4 USB Common PHY Control 1

BIOS: See 2.17.2.4.

Bits	Description
31:24	**DLLControl: DLL Control**. Read-write. Reset: A0h. Bit — Definition [1:0] — Reserved. [2] — DLL_EN_PFDphases. 1=Enable DLL phase-sampling based lock detection. [5:3] — DLL_cpump. DLL charge pump current control. [7:6] — DLL_VtoI. DLL gain control.
23:21	Reserved.
20	**PLLBypass: PLL Bypass**. Read-write. Reset: 0. 1=Enable USB Common PLL bypass.
19:16	**DutyAdj**. Read-write. Reset: 4h. Specifies CLK480 duty cycle control from 40-60% to 60-40%.
15:12	**PVI**. Read-write. Reset: 2h. Specifies PLL V-I Converter Control for common block PLL.
11:8	**IRefAdj**. Read-write. Reset: 1. Specifies internal reference bias adjustment for common block.
7:4	**XRefAdj**. Read-write. Reset: 1. Specifies external reference bias adjustment for common block.
3:0	**CPAdj**. Read-write. Reset: 4h. Charge pump setting for common block PLL.

EHCI1xD0 USB Common PHY Control 2

Bits	Description
31:16	Reserved.
15	**BgBypass.** Read-write. Reset: 0. Bandgap bypass.
14	Reserved.
13:12	**PllLockAdj[1:0]**. Read-write. Reset: 0. PLL lock detection tuning.
11	**UnlockSticky**. Read-only. Reset: X. Sticky unlock detection observability.
10	**UnlockReset**. Read-write. Reset: 0. Reset sticky un-lock detector.
9	**LockDetect**. Read-only. Reset: X. Lock detection observability.

8	**EnLockDetect**. Read-write. Reset: 0. Enable PLL lock detection.
7:4	**CBackUp.** Read-write. Reset: 0h. Common block backup.
3:0	**BgAdj.** Read-write. Reset: 6. BIOS: See 2.17.2.4. Bandgap voltage adjust.

EHCI1xD4 USB Common PHY Control 3

BIOS: See 2.17.2.4.

Bits	Description
31:8	Reserved.
7:6	**PllFilter**. Read-write. Reset: 0. PLL Filter tuning setting.
5:4	**PllDiv**. Read-write. Reset: 0. PLL divider setting.
3:2	**PllReg**. Read-write. Reset: 0. PLL Regulator tuning setting.
1	**CalEnable**. Read-write. Reset: 0. Calibration Enable.
0	**BgStart**. Read-write. Reset: 0. Bandgap startup tuning setting.

EHCI1xDC USB Battery Charger Enable

Bits	Description
31:4	Reserved.
3:0	**UsbBatteryChargerEnable**. Read-write. Reset: 0. One bit controls one respective port. For each bit, 1=Enable USB PHY battery charger function for the port.

3.26.4.2 USB 3.0 (xHCI)

The xHCI controller has a PCI configuration register space. The PCI configuration register space uses device 10h function 0 and is defined below as D10F0xXX.

3.26.4.2.1 Device 10h Function 0 (xHCI) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D10F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: Device ID**. Read-only. Reset: 7914h. Device Identifier.
15:0	**VendorID: Vendor ID**. Read-only. Reset: 1022h. Vendor Identifier.

D10F0x04 Status/Command

Bits	Description
31	**DetectedParityError: Detected Parity Error**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0.

30	**SignaledSystemError: Signaled System Error**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=System error signaled via SERR#.
29	**ReceivedMasterAbort: Received Master Abort**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0.
28	**ReceivedTargetAbort: Received Target Abort**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0.
27	**SignaledTargetAbort: Signaled Target Abort**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0.
26:25	**DEVSELTiming: DEVSEL Timing**. Value: 0.
24	**MasterDataParityError: Master Data Parity Error**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0.
23	**FastBack2BackCapable: Fast Back-to-Back Capable**. Value: 0.
22	Reserved.
21	**Capable66MHz: 66 MHz Capable**. Value: 0.
20	**CapabilitiesList: Capabilities List**. Value: 1. 1=Indicates that the value read at offset 34h is a pointer in Configuration Space to a linked list of new capabilities.
19	**InterruptStatus: Interrupt Status**. Read-only. Reset: 0. 1=An INTx interrupt message is pending in the device.
18:11	Reserved.
10	**InterruptDisable: Interrupt Disable**. Read-only. Reset: 0. 0=Enable the assertion of the device/function's INTx# signal. 1=Disable the assertion of the device/function's INTx# signal.
9	**FastBack2BackEnable: Fast Back-to-Back Enable**. Value: 0. 0=Only fast back-to-back transactions to the same agent are allowed.
8	**SERREnable: SERR# Enable**. Read-write. Reset: 0. 1=System error reporting enabled.
7	RAZ.
6	**ParityEnable: Parity Enable**. Read-write. Reset: 0. 0=The device sets its DetectedParityError when an error is detected, but continues normal operations without asserting PERR#. 1=The device must take its normal action when a parity error is detected.
5	**VgaPaletteRegisterAccesses: VGA palette register accesses**. Value: 0. 0=The device should treat palette write accesses like all other accesses.
4	**MemoryWriteandInvalidateCommand: Memory Write and Invalidate Command**. Value: 0. 0=Memory Write must be used.
3	**SpecialCycle: Special Cycle**. Value: 0. 0=No Special Cycle support.
2	**BusMaster: Bus Master**. Read-write. Reset: 0. 0=Disable the device from generating PCI accesses. 1=Allow the device to behave as a bus master.
1	**MemorySpaceAccesses: Memory Space Accesses**. Read-write. Reset: 0. 0=Disable the device response. 1=Allow the device to respond to Memory Space accesses.
0	**IOSpaceAccesses: IO Space Accesses**. Value: 0. 0=IO accesses targeting this device are not accepted.

D10F0x08 Revision ID / Class Code

Bits	Description
31:24	**BaseClass: Base Class**. Read-only. Reset: Ch. Base Class. Serial Bus Controller.
23:16	**SubClass: Sub Class**. Read-only. Reset: 3h. Sub Class. Universal Serial Bus Host Controller.

15:8	**ProgrammingInterface: Programming Interface**. Read-only. Reset: 30h. USB 3.0 Host Controller that conforms to xHC specification.
7:0	**RevisionID**. Read-only. Reset: 20h.

D10F0x0C Miscellaneous

Bits	Description
31:24	**Bist**. Value: 0. 00h=BIST is not supported.
23:16	**HeaderType: Header Type**. Value: 00h. 00h=The device does not have multiple functions.
15:8	**LatencyTimer: Latency Timer**. Value: 0.
7:0	**CacheLineSize: Cache Line Size**. Read-write. Reset: 0. This read/write field specifies the system cache line size in units of doublewords and must be initialized to 00h.

D10F0x10 Bar 0

Bits	Description
31:13	**Bar0**. Read-write. Reset: 0. Specifies Base Address[31:13].
12:4	Reserved.
3	**PM: Prefetch memory**. Value: 0. 0=Indicates that there is no support for prefetchable memory.
2:1	**Tp: Type**. Value: 10b. 10b=64-bit address.
0	**Ind: Indicator**. Value: 0. 0=Indicates that the operational registers of the device are mapped into memory space of the main memory of the PC host system.

D10F0x14 Bar 1

Bits	Description
31:0	**Bar1: Base Address High**. Read-write. Reset: 0. Specifies Base Address[63:32].

D10F0x2C Subsystem Vendor ID / Subsystem ID

Bits	Description
31:16	**SubsystemID: Subsystem ID**. Read-only. Reset: 7914h. Subsystem ID.
15:0	**SubsystemVendorID: Subsystem Vendor ID**. Read-only. Reset: 1022h. Subsystem Vendor ID.

D10F0x34 Capability Pointer

Bits	Description
7:0	**CapabilityPointer: Capability Pointer**. Read-only. Reset: 50h. Address of the 1st element of capability link.

D10F0x3C Interrupt Line

Bits	Description
31:16	Reserved.
15:8	**InterruptPin: Interrupt Pin**. Read-only. Reset: 1. Specifies which interrupt pin the device (or device function) uses. A value of 1 corresponds to INTA#.
7:0	**InterruptLine: Interrupt Line**. Read-write. Reset: 0. Specifies which input of the system interrupt controller(s) to which the device's interrupt pin is connected. The device itself does not use this value; rather it is used by device drivers and operating systems to determine priority and vector information.

D10F0x40 IDP Index Register

Bits	Description
31:13	Reserved.
12:2	**IDPIndex: IDP Index**. Read-only. Reset: 0. This register selects the doubleword offset of the memory mapped register to be accessed. All register accesses are at doubleword granularity.
1:0	Reserved.

D10F0x44 IDP Data Register

Bits	Description
31:0	**IDPData: IDP Data**. Read-only. Reset: 0. Specifies the data read from or written to the memory mapped register pointed to by D10F0x40. A physical register is not actually implemented as the data is actually stored in the memory mapped registers. The default value is the same as the default value of the register pointed to by D10F0x40.

D10F0x48 Indirect PCI Index Register

Bits	Description
31:30	**IndirectPCIIndex[31:30]: Indirect PCI Index**. Read-write. Reset: 0. Selects the PCI indirect space. This field is combined with IndirectPCIIndex[15:2] to select the PCI indirect register to be accessed. • IndirectPCIIndex[31:0] range for 3.26.4.2.1.1 [USB 3.0 125 MHz PCI Indirect Space]: 0000_FFFFh-0000_0000h. • IndirectPCIIndex[31:0] range for 3.26.4.2.1.2 [USB 3.0 60 MHz PCI Indirect Space]: 4000_FFFFh-4000_0000h. Bits / Definition / Bits / Definition 00b / Selects 125 MHz PCI indirect space / 10b / Reserved 01b / Selects 60 MHz PCI indirect space / 11b / Reserved
29:16	Reserved.
15:2	**IndirectPCIIndex[15:2]: Indirect PCI Index**. Read-write. Reset: 0. Selects the doubleword offset of the PCI indirect space register to be accessed. All register accesses are at doubleword granularity.
1:0	Reserved.

D10F0x4C Indirect PCI Data Register

Bits	Description
31:0	**IndirectPCIData: Indirect PCI Data**. Read-write. Reset: 0. Specifies the data written to/read from the PCI indirect space register pointed to by D10F0x48. A physical register is not actually implemented as the data is actually stored in the PCI indirect space registers. The default value is the same as the default value of the register pointed to by D10F0x48.

3.26.4.2.1.1 USB 3.0 125 MHz PCI Indirect Space

These registers are indirectly accessed through D10F0x48 and D10F0x4C.

D10F0x4C_x08 Port Disable Write Once

Bits	Description
31:4	Reserved.
3:0	**XhcPortDisableWriteOnce: xHC Port Disable Write Once**. Read; Write-once; Write-1-only. Reset: 0. This field is bit significant. Once written, the register can only be cleared by PciRst#. For each bit, 1=Disable the corresponding port. Bit Definition [0] port 0 [1] port 1 [2] port 2 [3] port 3 If (D10F0x4C_x08[XhcPortDisableWriteOnce] \| D10F0x4C_x0C[XhcPortDisableRW]), then the corresponding xHC port is disabled.

D10F0x4C_x0C Port Disable RW

Bits	Description
31:4	Reserved.
3:0	**XhcPortDisableRW: xHC Port Disable RW**. Read-write. Reset: 0. This field is bit significant. For each bit, 1=The corresponding port is disabled. Bit Definition [0] port 0 [1] port 1 [2] port 2 [3] port 3 If (D10F0x4C_x08[XhcPortDisableWriteOnce] \| D10F0x4C_x0C[XhcPortDisableRW]), then the corresponding xHC port is disabled.

D10F0x4C_x10 USB DCLK Event Counter 0

Bits	Description
31:0	**UsbDclkEventCnt0lo**. Read-only. Reset: 0. USB DClk Event Counter 0[31:0].

D10F0x4C_x14 USB DCLK Event Counter 1

Bits	Description
31:0	**UsbDclkEventCnt1Lo**. Read-only. Reset: 0. USB DClk Event Counter 1[31:0].

D10F0x4C_x18 USB DCLK Event Counter Select

Bits	Description
31:24	**UsbDclkEventCnt1Hi**. Read-only. Reset: 0. USB DClk Event Counter 1[39:32].
23:16	**UsbDclkEventCnt0Hi**. Read-only. Reset: 0. USB DClk Event Counter 0[39:32].
15:8	**UsbDclkEventSel1**. Read-write. Reset: 0. Select the event to be counted by USB DClk Event Counter 1.
7:0	**UsbDclkEventSel0**. Read-write. Reset: 0. Select the event to be counted by USB DClk Event Counter 0.

D10F0x4C_x1C USB DCLK Event Counter Control

Bits	Description
31:28	**Usb_DClk_EventCnt1SSport.** Read-write. Reset: 0. Select which SS port event source for event counter1. Bits — Definition 0h — SS port1 event source is selected. 1h — SS port2 event source is selected. Eh-2h — SS port <Usb_DClk_EventCnt1SSport> +1 event source is selected. Fh — SS port16 event source is selected. All SS port event sources (ssphyif event source: Event_2~Event_27, SS dma request/transfer length event source: Event_33, Event_34, …) share it.
27:24	**Usb_DClk_EventCnt0SSport.** Read-write. Reset: 0. Select which SS port event source for event counter0. Bits — Definition 0h — SS port1 event source is selected. 1h — SS port2 event source is selected. Eh-2h — SS port <Usb_DClk_EventCnt0SSport> +1 event source is selected. Fh — SS port16 event source is selected. All SS port event sources (ssphyif event source: Event_2~Event_27, SS dma request/transfer length event source: Event_33, Event_34, …) share it.
23:5	Reserved.
4	**Usb_DClk_EventCnt1Mode**. Read-write. Reset: 0. Select pulse mode or level mode for event counter1. 0=Level mode. 1=Pulse mode.
3	**Usb_DClk_EventCnt0Mode**. Read-write. Reset: 0. Select pulse mode or level mode for event counter0. 0=Level mode. 1=Pulse mode.
2	**UsbDclkEventCntShadow**. Read-write. Reset: 0. Transfer USB DClk event counter to shadow register.

1	**UsbDclkEventCntResetb**. Read-write. Reset: 1. Active low reset for USB DClk event counters.
0	**UsbDclkEventCntEn**. Read-write. Reset: 0. 1=Enable USB DClk event counters.

3.26.4.2.1.2 USB 3.0 60 MHz PCI Indirect Space

These registers are accessed through D10F0x48 and D10F0x4C.

D10F0x4C_x4000_0000 UTMI Control

Bits	Description
31:18	Reserved.
17	**VBusy**. Read-only. Reset: 0.
16:13	**PortNumber: Port Number**. Read-write. Reset: 0. Selects the corresponding port PHY or common block to load the VControl bits. Bits Definition Bits Definition 0000b Port 0 0011b Port 3 0001b Port 1 1110b-0100b Reserved 0010b Port 2 1111b Common block
12	**VLoadB**. Read-write. Reset: 1. Update PHY control mode (active low). 0=Load the new VControl value to PHY/common block. 1=Only load VControlModeSel value to PHY to select different PHY status group (see D10F0x4C_x4000_0004); VControl[6:0] value inside PHY won't get affected.
11	Reserved.
10:7	**VControlModeSel**. Read-write. Reset: 0. Selects the PHY control mode group. See: EHCI1xB4[VControlModeSel].
6:0	**VControl**. Read-write. Reset: 24h. Controls PHY setting.

D10F0x4C_x4000_0004 USB PHY Status

Bits	Description
31:24	**Port3PhyStatus**. Read-only; Updated-by-hardware. Reset: 21h. Specifies the PHY Status of Port 3. Use the D10F0x4C_x4000_0000[VControlModeSel] to select which group's status should be read back.
23:16	**Port2PhyStatus**. Read-only; Updated-by-hardware. Reset: 21h. PHY Status of Port 2. Use the D10F0x4C_x4000_0000[VControlModeSel] to select which group's status should be read back.
15:8	**Port1PhyStatus**. Read-only; Updated-by-hardware. Reset: 21h. Specifies the PHY Status of Port 1. Use the D10F0x4C_x4000_0000[VControlModeSel] to select which group's status should be read back.
7:0	**Port0PhyStatus**. Read-only; Updated-by-hardware. Reset: 21h. PHY Status of Port 0. Use the D10F0x4C_x4000_0000[VControlModeSel] to select which group's status should be read back.

D10F0x4C_x4000_0018 USB Common PHY Calibration and Control

Bits	Description
31	Reserved.

30:24	**CommonPhyCalBus**. Read-only; Updated-by-hardware. Reset: X. Specifies PHY common calibration bus.
23	Reserved.
22:20	**ComCalAmp**. Read-only; Updated-by-hardware. Reset: X. Specifies the amplitude calibration data from PHY.
19:18	Reserved.
17	**AddToCommonCalibration**. Read-write. Reset: 1. 1=The signed NewCalBus is added to the ComCalBus and returned to the PHY ports; Any overflow is clamped to all 1s; Any underflow is clamped to all 0s. 0=The signed NewCalBus replaces ComCalBus and returns to the PHY ports.
16	**UseCommonCalibration**. Read-write. Reset: 0. 1=The PHY's calibration value in ComCalBus is returned to the PHY ports. 0=The value after adjustment is returned to the PHY ports.
15:8	**NewCalBus[7:0]**. Read-write. Reset: 0. Specifies the new calibration bus signed value.
7	Reserved.
6:0	**ComCalBus**. Read-only; Updated-by-hardware. Reset: X. Specifies the common calibration bus value from PHY before adjustment.

D10F0x4C_x4000_001C USB Common PHY Control

BIOS: See 2.17.2.7 [xHC USB2.0 Common PHY Calibration]

Bits	Description
31:24	**DllControl: DLL Control**. Read-write. Reset: 90h. Specifies USB PHY DLL control. Bit — Definition [1:0] — Reserved. [2] — DLL_EN_PFDphases. 1=Enable DLL phase-sampling based lock detection. [5:3] — DLL_cpump. DLL charge pump current control. [7:6] — DLL_VtoI. DLL gain control.
23:21	Reserved.
20	**PllBypass: PLL Bypass**. Read-write. Reset: 0. 1=Enable USB Common PLL bypass.
19:16	**DutyAdj**. Read-write. Reset: 4h. Specifies CLK480 duty cycle control from 40-60% to 60-40%.
15:12	**PVI**. Read-write. Reset: 1h. Specifies PLL V-I converter control for common block PLL.
11:8	**IRefAdj**. Read-write. Reset: 2. Specifies internal reference bias adjustment for common block.
7:4	**XRefAdj**. Read-write. Reset: 2. Specifies external reference bias adjustment for common block.
3:0	**CPAdj**. Read-write. Reset: 1h. Specifies charge pump setting for common block PLL.

D10F0x4C_x4000_0020 HS Loopback Test

Bits	Description
31:20	Reserved.
19:12	**GoodReceivedPacketCount**. Read-only; Updated-by-hardware. Reset: 0. Specifies the number of good packets that the host controller received during the loopback test mode. These bits are cleared by programming EnableLoopBackTest = 0.
11	Reserved.

10	**LoopbackTestDone**. Read-only. Reset: 0. 1=Loopback test is done.
9	**LoopbackTestStatus**. Read-only. Reset: 0. 0=CRC error on loopback receiving data. 1=Good CRC on loopback receiving data.
8	**EnableLoopBackTest**. Read-write. Reset: 0. 1=Enable external USB port loopback test. The loop-back test is to set one port to TX mode (Test Packet mode) and one port to RX mode (Test SE0_NAK). See EHCI1x[70,6C,68,64][PortTestControl] for information on the tests.
7:4	Reserved.
3:0	**RxPktCnt: Received Packet Count**. Read-write. Reset: 0. This counter defines the number (as a power of 16) of RX data packets that should be checked for the loop back test.

D10F0x4C_x4000_0024 CL Loopback Control

Bits	Description
31	Reserved.
30:16	**LoopBackTestStatus**. Read-only; Updated-by-hardware. Reset: 0. This field is bit-significant. It specifies the loopback test status for the port under test. Software can only check these status bits when the LoopBackTestDone bit is set by the host controller. These bits can only be cleared when software clears the LoopBackTestStart bit. For each bit, 0=Fail. 1=Pass. Bit — Definition [0] — Port 0 [1] — Port 1 [4:2] — Reserved
15:8	**LoopBackTestData**. Read-write. Reset: 0. Specifies the 1-byte test data pattern for transmit; receive logic checks the received data to match with this data pattern.
7:4	**PortUnderTest**. Read-write. Reset: 0. Software selects the port under test through these bits. Bits — Definition 0001b-0000b — Port number <PortUnderTest> 1111b-0010b — Reserved
3	**LoopBackTestDone**. Read-only. Reset: 0. Host Controller sets the bit when loop back is done. The bit is cleared when software clears the LoopBackTestStart bit.
2:1	Reserved.
0	**LoopBackTestStart**. Read-write. Reset: 0. Software sets this bit to start the loopback test and clears this bit to clear out all the test status before starting the next loop. 1=Start the loopback test. 0=Clear LoopBackTestStatus and LoopBackTestDone.

D10F0x4C_x4000_0028 Misc Control

Bits	Description
31:12	Reserved.
11:8	**InterPacketGapAdjustCounter**. Read-write. Reset: 4h. Specifies the counter used to adjust the inter-packet gap for test packet.
7:3	Reserved.
2	**TestPacketInterPacketGapEnable**. Read-write. Reset: 1. 1=Enable inter-packet gap function for general test packet mode.

1	Reserved.
0	**U2IFPowerSavingEnable**. Read-write. Reset: 1. 1=Enable power saving clock gating; dynamic clock gating is enabled when U2IF trans_state is in idle state.

D10F0x4C_x4000_002C AMDU2IF_POWERUP_CHECK

Bits	Description
31:20	Reserved.
19:16	**StatusOfPowerUpStateCheck: Status of ports for PowerUp State Check**. Read-only. Reset: 0. Status bit to indicate the Power Up state auto-checking result from the individual port. Each bit maps to a corresponding port, from port-0 to port-3 (i.e., bit[16] for port-0, bit[17] for port-1, and so on). Bits Definition 0h PHY Power Up State checking is fail. 1h PHY Power Up State checking is good.
15:1	Reserved.
0	**EnablePHYPowerUpStateChecking: Enable PHY PowerUp State Checking**. Read-write. Reset: 0. Enable auto-checking on PHY Power Up State. There is built-in logic to check the PHY default Power Up state to detect manufacturing defects in the PHY macro.

D10F0x4C_x4000_0030 USB SCLK Event Counter 0

Bits	Description
31:0	**UsbSClkEventCnt0Lo**. Read-only. Reset: 0. USB SClk Event Counter 0[31:0].

D10F0x4C_x4000_0034 USB SCLK Event Counter 1

Bits	Description
31:0	**UsbSClkEventCnt1Lo**. Read-only. Reset: 0. USB SClk Event Counter 1[31:0].

D10F0x4C_x4000_0038 USB SCLK Event Counter Select

Bits	Description
31:24	**UsbSClkEventCnt1Hi**. Read-only. Reset: 0. USB SClk Event Counter 1[39:32].
23:16	**UsbSClkEventCnt0Hi**. Read-only. Reset: 0. USB SClk Event Counter 0[39:32].
15:8	**UsbSClkEventSel1**. Read-write. Reset: 0. Selects event to be counted by USB SClk Event Counter 1.
7:0	**UsbSClkEventSel0**. Read-write. Reset: 0. Selects event to be counted by USB SClk Event Counter 0.

D10F0x4C_x4000_003C USB SCLK Event Counter Control

Bits	Description
31:3	Reserved.

2	**UsbSClkEventCntShadow**. Read-write. Reset: 0. 1=Transfer USB SClk event counter to shadow register.
1	**UsbSClkEventCntResetB**. Read-write. Reset: 1. 0=Resets USB SClk event counters.
0	**UsbSClkEventCntEn**. Read-write. Reset: 0. 1=Enable USB SClk event counters.

D10F0x4C_x4000_0058 LPM Control

Bits	Description
31:10	Reserved.
9	**CfguDevRecoveryDis: Device Recovery Timer Disable**. Read-write. Reset: 0. 1=xHC does not wait for the device recovery time.
8:5	**CfguRWakeRsmDuration[3:0]: Remote Wake Resume Duration**. Read-write. Reset: 0. Software sets this field to indicate how long the xHC should stay in L1Resuming after remote wake is detected. The L1 Resume Timer loads this duration after remote wake is detected and HLE is 0. When the timer expires, the Root Hub state jumps to SENDEOR. Bits Definition Bits Definition 0000b 60 us 1011b-0011b 60 us + value * 75 us 0001b 135 us 1100b 960 us 0010b 210 us 1111b-1101b Reserved
4:0	**CfguL1ResidencyDuration: L1 Residency Duration**. Read-write. Reset: 0. Software sets this field to indicate the minimum duration that the xHC should stay in L1Suspend. The L1 Residency Timer loads this duration when LPM transaction ACK is received and HLE is 0. When the timer expires and there is either Hardware Initiated L1 Resume Request or Remote Wake detected, the Root Hub state jumps to L1Resuming. Bits Definition Bits Definition 00000b 50 us 11110b-00010b (value + 1)* 50 us 00001b 100 us 11111b 1.6 ms

D10F0x4C_x4000_0060 USB Common PHY Control 2

Bits	Description
31:4	Reserved.
3:0	**BgAdj**. Read-write. Reset: 6. Bandgap voltage adjust.

D10F0x4C_x4000_0064 USB Common PHY Control 3

Bits	Description
31:2	Reserved.
1	**CalEnable**. Read-write. Reset: 0. 1=Enable calibration.
0	Reserved.

D10F0x50 PME Capability

Bits	Description
31:27	**PmeSupport**. Read-only. Reset: 19h. Indicates the power states in which the function may assert PME#. For each bit, 0=The function is not capable of asserting the PME# signal while in that power state. Bit — Definition [0] — 1=PME# can be asserted from D0 [1] — 1=PME# can be asserted from D1 [2] — 1=PME# can be asserted from D2 [3] — 1=PME# can be asserted from D3hot [4] — 1=PME# can be asserted from D3cold
26	**D2Support**. Read-only. Reset: 0. 0=This function doesn't support the D2 Power Management State.
25	**D1Support**. Read-only. Reset: 0. 0=This function doesn't support the D1 Power Management State.
24:22	**AuxCurrent**. Read-only. Reset: 0. Specifies the 3.3 V auxiliary current requirements for the PCI function. Since D10F0x54[Data] is implemented by this function, reads of this field must return a value of 000b; D10F0x54[Data] takes precedence over this field for 3.3 V auxiliary current requirement reporting.
21	**DSI: Device Specific Initialization**. Read-only. Reset: 0. This bit indicates whether special initialization of this function is required (beyond the standard PCI configuration header) before the generic class device driver is able to use it.
20	Reserved.
19	**PmeClock: PME clock**. Read-only. Reset: 0. 0=Indicates that no PCI clock is required for the function to generate PME#.
18:16	**Version**. Read-only. Reset: 011b. 011b=Indicates that this function complies with Revision 1.2 of the PCI power management interface specification.
15:8	**NextItemPointer: Next Item Pointer**. Read-only. Reset: 70h. Points to the location of next item in the function's capability list.
7:0	**CapId**. Read-only. Reset: 1. The linked list item is for PCI Power Management.

D10F0x54 PME Control / Status

Bits	Description
31:24	**Data**. Read-only. Reset: 0. Reports the state dependent data requested by D10F0x54[DataSelect]. The value of this field is scaled by D10F0x54[DataScale].
23:16	Reserved.
15	**PmeStatus**. Read; Set-by-hardware; Write-1-to-clear. Cold reset: 0. BIOS: See 2.17.2.14. 1=The PME# signal is asserted, which is independent of PmeEn. 0=PME# is de-asserted.
14:13	**DataScale**. Read-only. Reset: 0. Indicates the scaling factor to be used when interpreting the value of D10F0x54[Data]. The value and meaning of this field varies depending on which data value has been selected by D10F0x54[DataSelect].
12:9	**DataSelect**. Read-only. Reset: 0. Selects which data is to be reported through D10F0x54[Data] and D10F0x54[DataScale].

8	**PmeEn**. Read-write. Cold reset: 0. 1=Enable the function to assert PME# if PmeStatus=1. 0=PME# assertion is disabled.
7:4	Reserved.
3	**NoSoftReset**. Read-only. Reset: 1. 1=Upon transitioning from D3hot to D0 because of PowerState, devices do not perform an internal reset; Configuration context is preserved; No additional operating system intervention is required to preserve configuration context beyond writing the PowerState bits. 0=Upon transitioning from D3hot to D0 via software control of the PowerState bits, devices do perform an internal reset; Configuration context is lost; Full reinitialization sequence is needed to return the device to D0 Initialized. Regardless of this bit, devices that transition from D3hot to D0 by a system or bus segment reset returns to the device state D0 uninitialized with only PME context preserved if PME is supported and enabled.
2	Reserved.
1:0	**PowerState**. Read-write. Reset: 0. This field determines the current power state of a function and sets the function into a new power state. If software attempts to write an unsupported, optional state to this field, the write operation must be completed normally on the bus; however, the data is discarded and no state change occurs. Bits Definition Bits Definition 00b D0 10b D2 01b D1 11b D3hot

D10F0x60 SBRN

Bits	Description
31:24	Reserved.
23:20	**DBESLD: Default Best Effort Service Latency Deep**. Read-only. Reset: 0h. Vendor Defined.
19:16	**DBESL: Default Best Effort Service Latency**. Read-only. Reset: 0h. Vendor Defined.
15:14	Reserved.
13:8	**FLAdj: Frame Length Timing Value.** Read-only. Reset: 20h. Each decimal value change to this register corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF microframe length) = 59488 + 16*FLAdj. Bits Frame Length (# HS bit times) 0 59488 1 59504 2 59520 30-3 58488 + (16 * <FLAdj>) 31 59984 32 60000 61-33 58488 + (16 * <FLAdj>) 62 60480 63 60496
7:0	**SBRN: Serial Bus Specification Release Number.** Value: 30h. Specifies the release of the Universal Serial Bus Specification with which this Universal Serial Bus Host Controller module is compliant.

D10F0x70 MSI Control

Bits	Description
31:25	Reserved.
24	**PVMC: Per-vector masking capable**. Read-only. Reset: 0. 1=The function supports MSI per-vector masking. 0=The function does not support MSI per-vector masking.
23	**C64: 64-bit Address Capable**. Read-only. Reset: 1. 1=The function is capable of sending a 64-bit message address. 0=The function is not capable of sending a 64-bit message address.
22:20	**MME: Multiple Message Enable**. Read-write. Reset: 0. System software writes to this field to indicate the number of allocated messages. Bits # of messages allocated: 000b 1; 001b 2; 010b 4; 011b 8; 100b 16; 101b 32; 111b-110b Reserved
19:17	**MMC: Multiple Message Capable**. Read-only. Reset: 011b. System software reads this field to determine the number of requested messages. 011b=8 messages requested.
16	**MSIE: MSI Enable**. Read-write. Reset: 0. 1=MSI is enabled and INTx# pins are not used to generate interrupts. 0=MSI is disabled and the traditional pins are used to request service.
15:8	**NextItemPointer: Next Item Pointer**. Read-only. Reset: 90h. Points to next the capability structure.
7:0	**Capability**. Read-only. Reset: 5h. MSI USB ID.

D10F0x74 MSI Address

Bits	Description
31:2	**MsiAddress: MSI Address**. Read-write. Reset: 0. Specifies the lower bits of system-specified message address [31:2]. Always doubleword aligned.
1:0	Reserved.

D10F0x78 MSI Upper Address

Bits	Description
31:0	**MsiUpperAddress: MSI Upper Address**. Read-write. Reset: 0. Specifies the system-specified message upper address[61:32].

D10F0x7C MSI Data

Bits	Description
15:0	**MsiData: MSI Data**. Read-write. Reset: 0. System-specified message.

D10F0x80 MSI Mask Bits

Bits	Description
31:0	**MaskBits: Mask Bits**. Read-write. Reset: 0. For each mask bit that is set, the function is prohibited from sending the associated message.

D10F0x90 MSI-X Control

Bits	Description
31	**MsiXEnable: MSI-X Enable**. Read-write. Reset: 0.
30	**FunctionMask: Function Mask**. Read-write. Reset: 0.
29:27	Reserved.
26:16	**TableSize: Table Size**. Read-only. Reset: 7h.
15:8	**NextItemPointer: Next Item Pointer**. Read-only. Reset: A0h.
7:0	**CapabilityID: Capability ID**. Read-only. Reset: 11h.

D10F0x94 MSI-X Table Offset/Table BIR

Bits	Description
31:3	**TableOffset: Table Offset**. Read-only. Reset: 200h. Used as an offset from the address contained by one of the function's Base Address registers to point to the base of the MSI-X Table. The lower 3 TableBIR bits are masked off (set to zero) by software to form a 32-bit, QWORD-aligned offset.
2:0	**TableBIR: Table BIR**. Read-only. Reset: 0. Indicates which Base Address register, starting at 10h in the configuration space, is used to map the function's MSI-X table into memory space. For a 64-bit Base Address register, the TableBIR indicates the lower DWORD. Bits / Base Address Register: 0h / 10h; 1h / 14h; 2h / 18h; 3h / 1Ch; 4h / 20h; 5h / 24h; 7h-6h / Reserved

D10F0x98 MSI-X PBA Offset/PBA BIR

Bits	Description
31:3	**PBAOffset: PBA Offset**. Read-only. Reset: 210h. Used as an offset from the address contained by one of the function's Base Address registers to point to the base of the MSI-X PBA. The lower 3 PBA BIR bits are masked off (set to zero) by software to form a 32-bit, QWORD-aligned offset.
2:0	**PBABIR: PBA BIR**. Read-only. Reset: 0. Indicates which Base Address register, starting at 10h in the configuration space, is used to map the functions MSI-X PBA into memory space. See: D10F0x94[TableBIR].

D10F0xA0 PCIe® Capability List

Bits	Description
31:30	Reserved.
29:25	**IntMsgNum: Interrupt Message Number**. Read-only. Reset: 0. This field indicates which MSI/MSI-X vector is used for the interrupt message generated in association with any of the status bits of this Capability structure.
24	**SlotImplemented: Slot Implemented**. Value: 0.
23:20	**DevicePortType: Device/Port Type**. Value: 9h. 9h=Indicates Root Complex Integrated Endpoint.
19:16	**CapabilityVersion: Capability Version**. Value: 2h. Indicates PCI-SIG defined PCI Express® Capability structure version number. 2h=PCI Express Base Rev 2.0.
15:8	**NextItemPointer: Next Item Pointer**. Value: 0. 00h=End of linked list capabilities.
7:0	**CapabilityID: Capability ID**. Read-only. Reset: 10h. PCI Express Capability.

D10F0xA4 Device Capability

Bits	Description
31:29	Reserved.
28	**FLRCapability: Function Level Reset Capability**. Read-only. Reset: 0. 0=Function Level Reset is not supported.
27:26	**SlotPowerLimitScale: Captured Slot Power Limit Scale (Upstream Ports only)**. Value: 0. Specifies the scale used for the SlotPowerLimitValue. Bits Definition 00b 1.0 x 01b 0.1 x 10b 0.01 x 11b0.001 x
25:18	**SlotPowerLimitValue: Captured Slot Power Limit Value (Upstream Ports only)**. Read-only. Reset: 0. In combination with SlotPowerLimitScale, specifies the upper limit on power supplied by slot. Power limit (in Watts)=SlotPowerLimitValue * SlotPowerLimitScale.
17:16	Reserved.
15	**RoleBasedErrorReport: Role-Based Error Reporting**. Value: 1. 1=Indicates that the function implements the functionality originally defined in the Error Reporting ECN for PCI Express® Base Specification, Revision 1.0a, and later incorporated into PCI Express Base Specification, Revision 1.1.
14:12	Reserved.
11:9	**EndpointL1AcceptableLatency: Endpoint L1 Acceptable Latency**. Value: 111b. This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. It is essentially an indirect measure of the Endpoint's internal buffering. Power management software uses the reported L1 Acceptable Latency number to compare against the L1 Exit Latencies reported (see below) by all components comprising the data path from this Endpoint to the Root Complex Root Port to determine whether ASPM L1 entry can be used with no loss of performance. 111b=No limit.

8:6	**EndpointL0sAcceptableLatency: Endpoint L0s Acceptable Latency**. Value: 111b. This field indicates the acceptable total latency that an Endpoint can withstand due to the transition from L0s state to the L0 state. It is essentially an indirect measure of the Endpoints internal buffering. Power management software uses the reported L0s Acceptable Latency number to compare against the L0s exit latencies reported by all components comprising the data path from this Endpoint to the Root Complex Root Port to determine whether ASPM L0s entry can be used with no loss of performance. 111b=No limit.
5	**ExtTagFieldSup: Extended Tag Field Supported**. Value: 0. This bit indicates the maximum supported size of the Tag field as a Requester. 0=Tag field size of 5-bits supported.
4:3	**PhantomFunctSup: Phantom Functions Supported**. Value: 0. 0=No Function Number bits are used for Phantom Functions.
2:0	**MaxPayloadSizeSup: Max Payload Size Supported**. Value: 0. This field indicates the maximum payload size that the function can support for PCIe® Transaction Layer Packets (TLP). Bits / Payload Size: 0h 128 bytes max payload size; 1h 256 bytes max payload size; 2h 512 bytes max payload size; 3h 1024 bytes max payload size; 4h 2048 bytes max payload size; 5h 4096 bytes max payload size; 7h-6h Reserved

D10F0xA8 Device Control/Status

Bits	Description
31:22	Reserved.
21	**TransPending: Transactions Pending**. Read-only. Reset: 0. 1=The function has issued Non-Posted Requests which have not been completed. 0=All outstanding Non-Posted Requests have completed or have been terminated by the Completion Timeout mechanism. This bit must also be cleared upon the completion of an Function Level Reset (FLR).
20	**AUXPowerDetected: AUX Power Detected**. Read-only. Reset: 0. 1=AUX power is detected by the function.
19	**UnsupReqDetected: Unsupported Request Detected**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=The function received an Unsupported Request. Errors are logged in this register regardless of UnsupReqRptEn.
18	**FatalErrDetected: Fatal Error Detected**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Fatal errors are detected. Errors are logged in this register regardless of FatalErRptEn.
17	**NonFatalErrDetected: Non-Fatal Error Detected**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Nonfatal errors are detected. Errors are logged in this register regardless of NonFatalErrRptEn.
16	**CorrErrDetected: Correctable Error Detected**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Correctable errors are detected. Errors are logged in this register regardless of CorrErrRptEn.
15	**InitiateFLR: Initiate Function Level Reset**. RAZ; Write-1-only. Reset: 0. FLR is not supported.
14:12	**MaxReadRequestSize**. Read-write. Reset: 010b. This field sets the maximum Read Request size for the function as a Requester. The function must not generate Read Requests with size exceeding the set value. 010b=512 bytes maximum Read Request size.
11	**EnNoSnoop: Enable No Snoop**. Read-write. Reset: 1. 1=Function is permitted to set the No Snoop bit in the Requester Attributes of transactions that it initiates and do not require hardware enforced cache coherency.

10	**AuxPowerPMEn: Auxiliary Power PM Enable**. Read-write. Cold reset: 0. 1=Enable a function to draw auxiliary (AUX) power independent of PME AUX power. Functions that require AUX power on legacy operating systems should continue to indicate PME AUX power requirements. AUX power is allocated as requested in D10F0x50[AuxCurrent] independent of D10F0x54[PmeEn]. Functions that consume AUX power must preserve the value of this sticky register when AUX power is available and preserve the value through Conventional Reset.
9	**PhantomFunctionsEn: Phantom Functions Enable**. Value: 0. 0=The Function is not allowed to use Phantom Functions.
8	**ExtTagFieldEn: Extended Tag Field Enable**. Value: 0. 0=Function is restricted to a 5-bit Tag field.
7:5	**MaxPayloadSize**. Read-write. Reset: 0. This field sets maximum TLP payload size for the function. as a receiver, the function must handle TLPs as large as the set value. As a transmitter, the Function must not generate TLPs exceeding the set value. Permissible values that can be programmed are indicated by D10F0xA4[MaxPayloadSizeSup]. Bits — Payload Size 0h — 128-byte max payload size 1h — 256-byte max payload size 2h — 512-byte max payload size 3h — 1024-byte max payload size 4h — 2048-byte max payload size 5h — 4096-byte max payload size 7h-6h — Reserved
4	**EnRelaxOrder: Enable Relaxed Ordering**. Read-write. Reset: 1. 1=Function is permitted to set the Relaxed Ordering bit in the Attributes field of transactions that it initiates and do not require strong write ordering.
3	**UnsupReqRptEn: Unsupported Request Reporting Enable**. Read-write. Reset: 0. This bit, in conjunction with other bits, controls the signaling of Unsupported Requests by sending Error Messages.
2	**FatalErrRptEn: Fatal Error Reporting Enable**. Read-write. Reset: 0. This bit, in conjunction with other bits, controls sending ERR_FATAL Messages
1	**NonFatalErrRptEn: Non-Fatal Error Reporting Enable**. Read-write. Reset: 0. This bit, in conjunction with other bits, controls sending ERR_NONFATAL Messages.
0	**CorrErrRptEn: Correctable Error Reporting Enable**. Read-write. Reset: 0. This bit, in conjunction with other bits, controls sending ERR_COR Messages.

D10F0xC4 Device Capabilities 2

Bits	Description
31:12	Reserved.
11	**LTRMechanismSupported: LTR Mechanism Supported**. Read-only. Reset: 1. 1=Latency Tolerance Reporting (LTR) mechanism capability is supported.
10:5	Reserved.
4	**CompletionTimeoutDisableSupported: Completion Timeout Disable Supported**. Read-only. Reset: 1. 1=The Completion Timeout Disable mechanism is supported.
3:0	**CompletionTimeoutRangesSupported: Completion Timeout Ranges Supported**. Read-only. Reset: 0. 0=Completion Timeout programmability mechanism is not supported.

D10F0xC8 Device Control/Status 2

Bits	Description
31:11	Reserved.
10	**LTRMechanismEnable: LTR Mechanism Enable**. Read-write. Reset: 0. 1=Enable the Latency Tolerance Reporting (LTR) mechanism.
9:5	Reserved.
4	**CompletionTimeoutDisable: Completion Timeout Disable**. Read-write. Reset: 0. 1=Disable the Completion Timeout mechanism.
3:0	**CompletionTimeoutValue: Completion Timeout Value**. Read-only. Reset: 0.

D10F0x100 LTR Extended Capability Header

Bits	Description
31:20	**NextCapabilityOffset: Next Capability Offset**. Read-only. Reset: 0. 0=End of capability list.
19:16	**CapabilityVersion: Capability Version**. Read-only. Reset: 1.
15:0	**PCIExpressExtendedCapabilityID: PCI Express® Extended Capability ID**. Read-only. Reset: 18h. Extended Capability ID for Latency Tolerance.

D10F0x104 Max Latency

Bits	Description
31:29	Reserved.
28:26	**MaxNoSnoopLatencyScale: Max No-Snoop Latency Scale**. Read-only. Reset: 0. This register provides a scale for the value contained within MaxNoSnoopLatencyValue. Encoding is the same as the Latency Scale fields in the Latency Tolerance Reporting (LTR) Message.
25:16	**MaxNoSnoopLatencyValue: Max No-Snoop Latency Value**. Read-only. Reset: 0. Along with MaxNoSnoopLatencyScale, this register specifies the maximum no-snoop latency that a device is permitted to request. Software should set this to the platform's maximum supported latency or less.
15:13	Reserved.
12:10	**MaxSnoopLatencyScale: Max Snoop Latency Scale**. Read-only. Reset: 0. This register provides a scale for the value contained within MaxSnoop LatencyValue. Encoding is the same as the Latency Scale fields in the LTR Message.
9:0	**MaxSnoopLatencyValue: Max Snoop Latency Value**. Read-only. Reset: 0. Along with MaxSnoopLatencyScale, this register specifies the maximum no-snoop latency that a device is permitted to request. Software should set this to the platform's maximum supported latency or less.

3.26.4.2.2 xHCI Memory Mapped IO Configuration Registers

3.26.4.2.2.1 USB xHCI Capability Registers

The base address of the xHCI Capability register space is specified by {D10F0x14[Bar1], D10F0x10[Bar0], 0h}.

XHCI_CAPx02 HC Interface Version

Bits	Description
15:0	**HCIVersion: Host Controller Interface Version Number**. Read-only. Reset: 0100h. Contains a BCD encoding of the xHCI specification revision number supported by this host controller. The most significant byte of this register represents a major revision and the least significant byte is the minor revision (e.g., 0100h corresponds to xHCI version 1.0). Writing to XHCI_PMx30[HCIVersion10En] updates this field.

3.26.4.2.3 xHCI Power Management Registers

xHCI power management registers are accessed through the AcpiMmio region. They range from FED8_0000h+1C00h to FED8_0000h+1CFFh. See PMx04[Mmio]. These registers reside in USB 3.0 and can hold their values during S3 or S5 state if the xHC host is programmed into D3 state before entering into S3/S5 state.

XHCI_PMx00 xHCI Config 0

Bits	Description
31	Reserved.
30	**FwPreloadComplete: Firmware Preload Complete**. Read-only; Updated-by-hardware. Reset: 0. 1=Firmware preload is complete. 0=FwPreloadStart is cleared by software.
29	**FwPreloadStart: Firmware Preload Start**. Read-write. Reset: 0. BIOS: See 2.17.2.13. When set from 0 to1, firmware preload is initiated. Upon completion of firmware preload (FwPreloadComplete == 1), this field should be programmed to 0.
28	**FwLoadMode: Firmware Load Mode.** Read-write. Reset: 1. Controls whether the firmware executes a bootstrap routine to load firmware into instruction ram, or whether instruction ram is to be preloaded. 0=xHCI executes bootstrap routine. 1=xHCI does not execute bootstrap routine.
27	Reserved.
26	**Xhc0BlmCgDis: xHC0 BLM Clock Gating Disable**. Read-write. Reset: 0. IF (U3CoreReset == 0) THEN clock gating takes effect. ENDIF. 1=Disable xHC0 BLM Clock Gating.
25	**AlGcgDis: A-Link Global Clock Gating Disable**. Read-write. Reset: 0. 1=Disable USB 3.0 A-Link Global Clock Gating.
24	**BlGcgDis: B-Link Global Clock Gating Disable**. Read-write. Reset: 0. 1=Disable USB 3.0 B-Link Global Clock Gating.
23:22	Reserved.
21	**XhcSmibEn: xHC SMIB Enable**. Read-write. Reset: 0. BIOS: 1. 1=Enable XHC_SMIB to ACPI controller.
20:17	Reserved.
16	**U3P_RESTORE_RESET**. Read-write. Reset: 1. 0=USB3.0 S0 restore and do reset. 1=keep alive. When xHC is back from D3 Cold, the BIOS sets this bit to 0 before doing shadow register and IRR/BTR preload. It is set to ‘1’ when both shadow register and IRR/BTR preload are finished.
15	**U3P_D3Cold_PWRDN**. Read-write. Reset: 0. 1=USB3.0 S0 logics power down. This bit is clear when xHC is back from D3 Cold and both shadow register and IRR/BTR preload are finished.
14	Reserved.

13:12	Reserved.
11	**Xhc0Reset: xHC0 Reset**. Read-write. Reset: 0. 1=Reset xHC0 core logic.
10	**U3CoreReset: USB3 Core Reset**. Read-write. Reset: 1. BIOS: See 2.17.2.13. 1=Reset the USB3.0/xHCI (xHC0) core logic. 0=xHC clock gating enable.
9	**U3pPhyReset: USB3 PHY Reset**. Read-write. Reset: 1. 1=Reset SSPHYIF power control logic in 125 MHz clock domain.
8	**U3pPllReset: USB3 PHY PLL Reset**. Read-write. Reset: 1. 1=Reset USB3.0 PHY PLL and SSPHYIF power control logic in LFPS clock domain.
7	**U3pLock: USB3 PHY PLL Lock**. Read-only. Reset: 0. 1=USB3.0 PHY PLL is locked.
6:1	Reserved.
0	**Xhci0Enable**. Read-write. Reset: 0. BIOS: See 2.17.2.3. 1=Enable xHCI0.

XHCI_PMx04 xHCI Firmware Addr 0

Bits	Description
31	**XhciFwPreloadType**. Read-write. Reset: 0. BIOS: See 2.17.2.13. Controls the destination of the pre-load operation. 0=Instruction RAM. 1=Boot RAM.
30:17	Reserved.
16:0	**XhciFwRomAddr**. Read-write. Reset: 0. BIOS: See 2.17.2.13. Specifies the address of the first byte of application firmware in external ROM.

XHCI_PMx08 xHCI Firmware Addr 1

Bits	Description
31:16	**XhciFwSize: xHCI Firmware Size.** Read-write. Reset: 0. BIOS: See 2.17.2.13. Specifies the size of the application firmware in the external ROM.
15:0	**XhciFwRamAddr: xHCI Firmware Start Address**. Read-write. Reset: 0. BIOS: See 2.17.2.13. Specifies the address where the first byte of application firmware is to be stored in boot RAM or instruction RAM.

XHCI_PMx10 xHCI Memory Config

Bits	Description
31:27	Reserved.
26	**MSE_RESYNC_EN**. Read-write. Reset: 0. BIOS: 1. Enable cur_uframe_tag rollover during resync. 1=If SI_ISO_MSE_ENHANCE_EN(0x10[23]) == 1, always support cur_uframe_tag rollover during resync. Otherwise, MSE_RESYNC_EN is the enable bit for cur_uframe_tag rollover during resync.
25	**TRB_EMPTY_NO_MSE_EN**. Read-write. Reset: 0. BIOS: 1. To enable no MSE report when ring underrun. 1=When a ring underrun is encountered, do not report MSE ERROR during resync.
24	**SI_INT_ENHANCE_EN**. Read-write. Reset: 0. BIOS: 1. Enable Service Interval enhancement for Interrupt. 0=xHC max supported Service Interval is 32ms for INT. 1=xHC would support full range of INT Service Interval value defined by USB and xHCI spec.

23	**SI_ISO_MSE_ENHANCE_EN**. Read-write. Reset: 0. BIOS: 1. Enable Service Interval enhancement for Isoch. 0=xHC max supported Service Interval is 4ms for ISO. MSE logic only supports up to 4ms service interval period. 1=xHC would support full range of ISO Service Interval value defined by USB and xHCI spec. Enable MSE enhancement logic to support >4ms service interval time. If SI_ISO_ENHANCE_EN is disabled, this bit does not take effect.
22:16	Reserved.
15:8	**CcuMode: Clock Control Mode.** Read-write. Reset: 3Fh. BIOS: 3Dh. This field contains a set of bits which, when set, enable individual features of the clock control unit. Bit — Definition [0] 1=Enable power control mode. [1] 1=Enable SuperSpeed remote wake mode. [2] 1=Enable PCLK control mode. [3] 1=Enable Memory Sleep Sequencing. [4] 1=Enable Stop CPU mode. [5] 1=Enable firmware clock control to start firmware after PCI reset is released. [7:6] Reserved.
7:2	Reserved.
1	**XhcNCpuMemSlpDis: Non-CPU Memory Sleep Disable**. Read-write. Reset: 0. Controls whether the sleep feature is disabled on xHCI Non-Cpu internal SRAMs. 0=Do not disable sleep. 1=Disable sleep.
0	**XhcCpuMemSlpDis: CPU Memory Sleep Disable.** Read-write. Reset: 0. Controls whether the sleep feature is disabled on xHCI Embedded CPU internal SRAMs. 0=Do not disable sleep. 1=Disable sleep.

XHCI_PMx14 PLL Control

Bits	Description
31:0	Reserved.

XHCI_PMx18 USB2.0 Link State

Bits	Description
31:27	Reserved.
26	**AOAC_PWRSAVING_EN**. Read-write. Reset: 0. BIOS: 1. AOAC Power Saving. 0=Disable. 1=Enable.
25	**LPC_FIFO_IMP_EN25**. Read-write. Reset: 0. BIOS: 1. LPC_XHC FIFO to shrink the preload time for AOAC. 0=Normal speed for preload time. 1=Enable the improvement and shrink the time.
24	**AOAC_SCRATCHPAD_EN**. Read-write. Reset: 0. BIOS: 1. Scratchpad enable for firmware in AOAC. 0=Disable. 1=Enable.
23:13	Reserved.
12	**BW_BMEIntfB2BEnable2**. Read-write. Reset: 0. BIOS: 1. Backward compatibility bit to disable BME interface back to back write transfers. 0=Disable. 1=Enable.
11	**BW_BMEIntfB2BEnable1**. Read-write. Reset: 0. BIOS: 1. Backward compatibility bit to disable BME interface back to back write transfers. 0=Disable. 1=Enable.

10	**BW_MORIntfB2BEnable2**. Read-write. Reset: 0. BIOS: 1. Backward compatibility bit to disable MOR interface back to back read transfers. 0=Disable. 1=Enalbe.
9	**BW_MORIntfB2BEnable1**. Read-write. Reset: 0. BIOS: 1. Backward compatibility bit to disable MOR interface back to back read transfers. 0=Disable. 1=Enable.
8	**BW_RPIPELINE_Enable**. Read-write. Reset: 0. BIOS: 1. DMA Read pipeline mode enable. 0=Disable pipeline. 1=Enable pipeline.
7:0	**U2IFInactSel: U2IF Inactive State Selection**. Read-write. Reset: A3h. Selects the USB2 link states in which the SuperSpeed PHY PLL may be shutdown. Bits Definition Bits Definition 0 Disabled 5h U3 1h RxDetect 6h Resume 2h Polling 7h Testing 3h Reset FFh-08h Reserved 4h U2

XHCI_PMx20 USB2.0 Wake Control

Bits	Description
31:26	Reserved.
25:24	**SysBW: System Bandwidth**. Read-write. Reset: 0. BIOS: 1. This field is used to provide information to the controller about the available system bandwidth.
23:22	**U2SE1DisconMode: xHC USB2 Interface SE1 Disconnect Mode.** Read-write. Reset: 1. 0=Take SE1 > 2.5 us as a disconnect. 1=Do not take SE1 as a disconnect.
21:0	Reserved.

XHCI_PMx2C xHCI Misc 2

Bits	Description
31:21	Reserved.
20	**PED_MODE**. Read-write. Reset: 0. BIOS: 1. Use this bit to control the setting of PED. 0=Set PED when link goes to U0. 1=Set PED when the downstream configuration successful.
19	**DCP_PP_PED_MODE**. Read-write. Reset: 0. BIOS: 1. The mode bit to select if PORTSC.PP and PORTSC.PED could affect LTSSM when DbC owns a port. 1=Does not affect LTSSM. 0=Affects LTSSM.
18	**D3_LATCH_OCA_PLS_MODE**. Read-write. Reset: 0. BIOS: 1. The bit is the mode bit for PLS when BLOCK_D3_HOT_P3_EXIT_EN is enabled. 0=Report PLS according to LTSSM when OCA occurs in D3 Hot. 1=Report PLS as SS.Disabed when OCA occurs in D3 Hot.
17	**BLOCK_D3_HOT_P3_EXIT_MODE**. Read-write. Reset: 0. The bit is the mode bit for blocked event when BLOCK_D3_HOT_P3_EXIT_EN is enabled. 0=Block any P3 exit event. 1=Block over current event only.
16	**BLOCK_D3_HOT_P3_EXIT_EN**. Read-write. Reset: 0. BIOS: 1. The bit is used to enable blocking P3 exit for LTSSM U3 state or Rx.Detect. Quiet state when D3 Hot. 0=Disabe. 1=Enable.

15	**DCP_EPST_HALTED_EN**. Read-write. Reset: 0. BIOS: 1. The bit enables EPState update to Halted and triggers EP context update when debug target driver write HIT/HOT to 1. 0=Disable. 1=Enable.
14	**DCP_HIT_HOT_CLR_MODE**. Read-write. Reset: 0. BIOS: 1. The bit to select HIT/HOT clear mode when detecting fetch TRB error. 0=HIT/HOT is cleared when get ClearFeature(ENDPOINT_HALT) request. 1=HIT/HOT is cleared next time when fetch new TRB.
13	**DCP_HIT_HOT_RW1S_MODE**. Read-write. Reset: 0. BIOS: 1. The bit to select HIT/HOT register attribute. 0=RW. 1=RW1S.
12	**DCP_DEVFMWK_EN**. Read-write. Reset: 0. BIOS: 1. This bit to enable the enhancement for DbC device framework. With the enhancement, the U1/U2 can only be enabled when in Configured state. 0=Disable. 1=Enable.
11	**DCP_DCR_EN**. Read-write. Reset: 0. BIOS: 1. This bit to enable the DCR enhancement of DbC. With the enhancement, the DCR is set to 0 when device state goes to error state or port is disabled. 0=Disable. 1=Enable.
10	**DCP_EPST_STOPPED_EN**. Read-write. Reset: 0. BIOS: 1. The bit enables EPState update to Stopped and triggers EP context update when debug target was in Halted and gets a ClearFeature(ENDPOINT_HALT) request from host machine. 0=Disable. 1=Enable.
9	**CL_CDR_SYNC_MODE**. Read-write. Reset: 0. BIOS: 1. This bit selects the sync type for the CV_CDR. 0=Use two FFs to sync logic. 1=Use the sync_cell to sync logic.
8	**CL_RXV_SYNC_MODE**. Read-write. Reset: 0. BIOS: 1. This bit selects the sync type for the CV_RXV. 0=Use two FFs to sync logic. 1=Use the sync_cell to sync logic.
7	**CL_FLT_SYNC_MODE**. Read-write. Reset: 0. BIOS: 1. This bit selects the sync type for the CV_FLT. 0=Use two FFs to sync logic. 1=Use the sync_cell to sync logic.
6	**FSE_TERMI_ED_FSE_MODE**. Read-write. Reset: 0. BIOS: 1. The mode select bit when send FSE based on terminating event data TRB. 0=Send ED FSE. 1=Send normal FSE.
5	**ZCAL_ENHANCE_DIS**. Read-write. Reset: 0. Enable pipeline. Calibration module enhancement (put registers with default value). 1=Disable the enhancement. 0=Enable the enhancement.
4	**XHC_PME_MODE_SEL**. Read-write. Reset: 0. BIOS: 1. Controls how to generate PME relying on PORTSC. 0=PORTSC reflects to PME Status continuously. 1=PORTSC reflects to PME Status on actual shutdown. Default mode, the PORTSC reflects to PME Status continuously.
3	**XHC_PME_enhancement_EN**. Read-write. Reset: 0. BIOS: 1. Controls the manner in which XHC generates the PME Status. 0=Disable. 1=Enable.
2	**UTMI_RXACTIVE_MASK_EN.** Read-write. Reset: 0. BIOS: 1. The bit to enable to mask RXACTIVE with ROOTHUB_STATE. 1=Enable to mask RXACTIVE with ROOTHUB_STATE. 0=Disable to mask RXACTIVE with ROOTHUB_STATE.
1	**FW_MSK_DBACTIVE_EN.** Read-write. Reset: 0. BIOS: 1. The bit to enable firmware to mask door bell active to hardware schedule assistant logic.1=Enable firmware mask door bell active to scheduler. 0=Disable firmware mask door bell active to scheduler.
0	**FSE_EDTD_EVT_SEL**. Read-write. Reset: 0. BIOS: 1. The mode selection bit for the event report behavior when executing stop endpoint command and current dequeue pointer is pointing to an Event Data TD. 1=Report transfer event and FSE event. 0=Only report FSE event.

XHCI_PMx30 xHCI 1.0 Enable

Bits	Description
31:16	Reserved.
15	**HCIVersion10En**. Read-write. Reset: 1. Controls the host controller interface version number reported in XHCI_CAPx02. 0=HCI Version is 0096h. 1=HCI Version is 0100h.
14	**MSEFrameIdEn: Missed Server Error Frame ID Enable**. Read-write. Reset: 1. 1=Enable a clarification in xHCI 1.0 specifying the conditions under which a Missed Service Error should be detected based on Frame ID. 0=Disable.
13:12	Reserved.
11	**SoftRetryEn: Soft Retry Enable**. Read-write. Reset: 1. 0=Do not support soft retry. 1=Support soft retry.
10	**SkipMsIocEvtEn: Skip Missed Service IOC Event Enable**. Read-write. Reset: 1. 1=Enable the parsing of TRBs during re-sync of an ISO pipe following detection of a Missed Service Error. 0=Disable. Always enable this bit when SkipTrbIocEvtEn == 1.
9	**EpStateUpdateChangeEn: EPState Update Change Enable**. Read-write. Reset: 1. 1=Enable a clarification in xHCI 1.0 specifying that when an endpoint transitions from Stopped to Running due to a doorbell ring, the EP State shall be updated to Running before any Transfer Events are generated. 0=Disable.
8	**CASEn: Cold Attach Status Enable**. Read-write. Reset: 1. 1=Enable the xHCI 1.0 feature Cold Attach Status flag. 0=Disable.
7	**SbdCapObsolete: SBD Reporting Capability Obsolete**. Read-write. Reset: 1. 1=Enable xHCI 1.0 feature which makes Secondary Bandwidth Domain Reporting mandatory. 0=Disable.
6	**SkipTrbIocEvtEn: Skip TRB IOC Event Enable**. Read-write. Reset: 1. 1=Enable xHCI 1.0 features related to the parsing of TRBs and generation of events while skipping over TRBs due to errors and short packets. 0=Disable.
5:4	Reserved.
3	**SwLpmEn: Software LPM Enable**. Read-write. Reset: 1. 0=Disable software controlled LPM feature. 1=Enable software controlled LPM feature.
2	**FseEn: FSE Enable**. Read-write. Reset: 1. 0=Disable FSE feature. 1=Enable FSE feature.
1	**IntBlockEn: Interrupt Blocking Enable**. Read-write. Reset: 1. 0=Disable interrupt blocking. 1=Enable interrupt blocking.
0	**Xhci10En: xHCI 10 Enable**. Read-write. Reset: 0. Global enable bit for xHCI 1.0 features. All xHCI 1.0 features may be enabled by setting this bit, or can be individually enabled by setting the individual feature enables. 1=Enable all xHCI 1.0 features.

XHCI_PMx48 SSPHY ACPI Indirect Index

Bits	Description
31:16	Reserved.
15:2	**IndirectIndex: Indirect Index**. Read-write. Reset: 00h.
1:0	Reserved.

XHCI_PMx4C SSPHY ACPI IndirectData

Bits	Description
31:0	**IndirectData: Indirect Data.** Read-write. Reset: 0000_0000h.

3.26.4.2.3.1 xHCI SSPHY 60 MHz ACPI Indirect Space

XHCI_PMx4C_x30 Over current Control

Bits	Description
31:16	**Reserved_ctl_4: Reserved control 4**. Read-write. Reset: FFFFh.
15:12	**Over_currect_ctl_3: over current control 3**. Read-write. Reset: Fh. The over current pin mapping for port-3 and Port7. See Over_currect_ctl_0.
11:8	**Over_currect_ctl_2: over current control 2**. Read-write. Reset: Fh. The over current pin mapping for port-2 and Port6. See Over_currect_ctl_0.
7:4	**Over_currect_ctl_1: over current control 1**. Read-write. Reset: Fh.The over current pin mapping for port-1 and Port5. See Over_currect_ctl_0.
3:0	**Over_currect_ctl_0: over current control 0**. Read-write. Reset: Fh.The over current pin mapping for port-0 and Port4. Bits — Definition 0000b — USB_OC0#. 0001b — USB_OC1#. 0010b — USB_OC2#. 0011b — USB_OC3#. 0100b — USB_OC4#. 0101b — USB_OC5#. 0110b — USB_OC6#. 0111b — USB_OC7#. 1111b-1000b — Disables the over current function.

XHCI_PMx4C_x2[C,8,4,0]0 SSPHY Port[3:0] Test Control 0

Bits	Description
31:3	Reserved.
2:0	**RxLfpsDetTh[2:0]**. Read-write. Reset: 4h. Specifies the port Low Frequency Periodic Signaling (LFPS) differential detection threshold adjustment. If the platform nominal power supply is 0.95v, then set [2:0] = 4h. If the platform nominal power supply is 1.05v, then set [2:0] = 3h.

XHCI_PMx8C SSPHY Common Control 0

Bits	Description
31:25	Reserved.

24	**CrPllCalibEn**. Read-write. Reset: 0h. BIOS: See 2.17.2.9. 1=Auto calibration is enabled; use calibrated value as the VCO setting. 0=Auto calibration is disabled; Use CP_PLL_VCO_TUNE[3:0] as the VCO setting if auto calibration has not been enabled; Use previously calibrated value as the VCO setting if auto calibration has been enabled once.
23:20	Reserved.
19:16	**PllVcoTune[3:0]**. Read-write. Reset: 0h. BIOS: See 2.17.2.9. Specifies PLL VCO range initial setting (in Thermometer code). Valid settings are 0000b, 0001b, 0011b, 0111b, and 1111b. Programming invalid values may result in unpredictable behavior.
15:0	Reserved.

XHCI_PMxA0 SPI BAR0

Bits	Description
31:16	**XhcSpiDataType0Size**. Read-write. Reset: 0. Type 0 data size.
15:0	**XhcSpiDataType0Addr**. Read-write. Reset: 0. Type0 data ROM address.

XHCI_PMxA4 SPI BAR1

Bits	Description
31:16	**XhcSpiDataType1Size**. Read-write. Reset: 0. Type 1 data size.
15:0	**XhcSpiDataType1Addr**. Read-write. Reset: 0. Type 1 data ROM address.

XHCI_PMxA8 SPI BAR2

Bits	Description
31:16	**XhcSpiDataType2Size**. Read-write. Reset: 0. Type 2 data size.
15:0	**XhcSpiDataType2Addr**. Read-write. Reset: 0. Type 2 data ROM address.

XHCI_PMxAC SPI BAR3

Bits	Description
31:16	**XhcSpiDataType3Size**. Read-write. Reset: 0. Type 3 data size.
15:0	**XhcSpiDataType3Addr**. Read-write. Reset: 0. Type 3 data ROM address.

XHCI_PMxB0 SPI Valid Base

Bits	Description
31:26	**SpiBase3**. Read-write. Reset: 0. Specifies the offset in SPI data block of type 3 data.
25	Reserved.
24	**SpiBar3Vld: SPI BAR3 Valid**. Read-write. Reset: 0. 1=SPI_BAR4 register is valid.
23:18	**SpiBase2**. Read-write. Reset: 0. Specifies the offset in SPI data block of type 2 data.

17	Reserved.
16	**SpiBar2Vld: SPI BAR2 Valid**. Read-write. Reset: 0. 1=SPI_BAR2 register is valid.
15:10	**SpiBase1**. Read-write. Reset: 0. Specifies the offset in SPI data block of type 1 data.
9	Reserved.
8	**SpiBar1Vld: SPI BAR1 Valid**. Read-write. Reset: 0. 1=SPI_BAR1 register is valid.
7:2	**SpiBase0**. Read-write. Reset: 0. Specifies the offset in SPI data block of type 0 data.
1	Reserved.
0	**SpiBar0Vld: SPI BAR0 Valid**. Read-write. Reset: 0. 1=SPI_BAR0 register is valid.

XHCI_PMxB4 SPI Misc

Bits	Description
31:16	Reserved.
15:0	**XhcSpiFwId: SPI Firmware ID**. Read-write. Reset: 0. Firmware Version ID.

XHCI_PMxB8 FW_DMA_ADDR_LOW

Bits	Description
31:0	**FW_DMA_ADDR_LOW**. Read-write. Reset: 0. When xHC firmware is in main memory, system software sets the lower 32 bits of begin address for xHC firmware inside main memory.

XHCI_PMxBC FW_DMA_ADDR_HIGH

Bits	Description
31:0	**FW_DMA_ADDR_HIGH**. Read-write. Reset: 0. When xHC firmware is in main memory, system software sets the upper 32 bits of begin address for xHC firmware inside main memory.

XHCI_PMx[FC:C0] SPI Data Block N

Bits	Description
31:0	**XhcSpiDataBlock**. Read-write. Reset: 0. SPI Data Block.

3.26.4.3 USB Legacy Registers

This register space is used to provide USB legacy support. Each of the registers is located on a 32-bit boundary. The offsets of the registers are relative to the AcpiMMio base (FED8_0000h). Three of the operational registers (HCEx4C, HCEx44 and HCEx48) are accessible at IO060 and IO064 when emulation is enabled. Reads and writes to the registers using IO addresses have side effects as outlined in the table below.

Table 275: USB Legacy Registers and IO Ports

IOAddress	Cycle Type	Registers Contents Accessed/Modified	Side Effects
IO060	IN	HCEx48 [HCE Output]	IN from port 60h sets HCEx4C[OutputFull] to 0.
IO060	OUT	HCEx44 [HCE Input]	OUT to port 60h sets HCEx4C[InputFull] to 1 and HCEx4C[CmdData] to 0.
IO064	IN	HCEx4C [HCE Status]	IN from port 64h returns current value of HCEx4C [HCE Status] with no other side effect.
IO064	OUT	HCEx44 [HCE Input]	OUT to port 64h sets HCEx4C[InputFull] to 0 and HCEx4C[CmdData] to 1.

HCEx40 HCE Control

Reset: 0000_0000h.

Bits	Description
31:9	Reserved.
8	**A20State**. Read-write. Indicates current state of Gate A20 on keyboard controller. Used to compare against value written to IO port 60h when GateA20Sequence is active.
7	**IRQ12Active**. Read; Write-1-to-clear. 1=A positive transition on IRQ12 from the keyboard controller has occurred.
6	**IRQ1Active**. Read; Write-1-to-clear. 1=A positive transition on IRQ1 from the keyboard controller has occurred.
5	**GateA20Sequence**. Read-write. Set by HC when a data value of D1h is written to IO port 64h. Cleared by HC on write to IO port 64h of any value other than D1h.
4	**ExternalIRQEn**. Read-write. 1=Enable IRQ1 and IRQ12 from the keyboard controller to cause an emulation interrupt. The function controlled by this bit is independent of the setting of the Emulation-Enable bit in this register.
3	**IRQEn**. Read-write. 1=Enable the HC generation of IRQ1 or IRQ12 as long as the (HCEx4C[OutputFull] ==1). IF (HCEx4C[AuxOutputFull] == 0) THEN IRQ1 is generated. ELSE IRQ12 is generated.
2	**CharacterPending**. Read-write. 1=An emulation interrupt is generated when HCEx4C[OutputFull]=0.
1	**EmulationInterrupt**. Read-only. This bit is a static decode of the emulation interrupt condition.
0	**EmulationEnable**. Read-write. 1=Enable HC legacy keyboard and mouse emulation. The HC decodes accesses to IO060 and IO064 and generates IRQ1 and/or IRQ12 when appropriate. Additionally, the HC generate s an emulation interrupt at appropriate times to invoke the emulation software.

HCEx44HCE Input

IO data that is written to ports 60h and 64h is captured in this register when emulation is enabled. This register may be read or written directly by accessing it with its memory address in the Host Controller's operational register space. When accessed directly with a memory cycle, reads and writes of this register have no side

effects.

Bits	Description
31:8	Reserved.
7:0	**InputData**. Read-write. Reset: 0. This register holds data that is written to IO ports 60h and 64h.

HCEx48 HCE Output

The data placed in this register by the emulation software is returned when IO port 60h is read and emulation is enabled. On a read of this location, HCEx4C[OutputFull] is set to 0.

Bits	Description
31:8	Reserved.
7:0	**OutputData**. Read-write. Reset: 0. This register hosts data that is returned when an IO read of port 60h is performed by application software.

HCEx4C HCE Status

Reset: 0000_0000h. The contents of the HceStatus Register are returned on an IO Read of port 64h when emulation is enabled. Reads and writes of port 60h and writes to port 64h can cause changes in this register. Emulation software can directly access this register through its memory address in the Host Controller's operational register space. Accessing this register through its memory address produces no side effects.

Bits	Description
31:8	Reserved.
7	**Parity**. Read-write; Updated-by-hardware. 1=Indicates parity error on keyboard/mouse data.
6	**TimeOut**. Read-write; Updated-by-hardware. 1=Indicates a timeout.
5	**AuxOutputFull**. Read-write; Updated-by-hardware. IRQ12 is asserted whenever this bit is set to 1 and OutputFull is set to 1 and the IRQEn bit is set.
4	**InhibitSwitch**. Read-write; Updated-by-hardware. This bit reflects the state of the keyboard inhibit switch and is set if the keyboard is NOT inhibited.
3	**CmdData**. Read-write; Updated-by-hardware. The HC sets this bit to 0 on an IO write to port 60h and to 1 on an IO write to port 64h.
2	**Flag**. Read-write; Updated-by-hardware. Nominally used as a system flag by software to indicate a warm or cold boot.
1	**InputFull**. Read-write; Updated-by-hardware. Except for the case of a Gate A20 sequence, this bit is set to 1 on an IO write to address 60h or 64h. While this bit is set to 1 and emulation is enabled, an emulation interrupt condition exists.
0	**OutputFull**. Read-write; Updated-by-hardware. The HC sets this bit to 0 on a read of IO port 60h. If HCEx40[IRQEn] == 1 and AuxOutputFull == 0, then an IRQ1 is generated as long as this bit is set to 1. If HCEx40[IRQEn] == 1 and AuxOutputFull == 1, then an IRQ12 is generated as long as this bit is set to 1. While this bit is 0 and HCEx40[CharacterPending] == 1, an emulation interrupt condition exists.

HCEx50HCE IntrEn

Bits	Description
31:11	Reserved.
10	**A20State_inv**. Read-write. Reset: 0. 1=Set to 1 to use inverted version of A20State.
9:8	**SOFSrc**. Read-write. Reset: 00b. Indicates which HCI_MSofN source is used in USB Legacy. 0=HCI_MSofN from EHCI1 (Dev12, func0); 1=HCI_MSofN from EHCI2 (Dev12, func0); 2=HCI_MSofN from EHCI3 (Dev12, func0).
7:5	**SOFCounter**. Read-write. Reset: 101b. Specifies the amount of SOF number that emulation interrupt has to wait before SMI is sent.
4	**EmulationSmiEn**. Read-write. Reset: 0. 1=Gnerate SMI if emulation interrupt happens.
3	**BlockIRQ12**. Read-write. Reset: 0. 1=Block IRQ12.
2	**BlockIRQ1**. Read-write. Reset: 0. 1=Block IRQ1.
1	**EnIRQ12**. Read-write. Reset: 1. 1=Enable IRQ12 active.
0	**EnIRQ1**. Read-write. Reset: 1. 1=Enable IRQ1 active.

3.26.5 Secure Digital (SD) Controller

3.26.5.1 Device 14h Function 7 Configuration Registers (SD)

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D14F7x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: Device ID**. Value: 7906h.
15:0	**VendorID: Vendor ID**. Value: 1022h. Specifies a unique 16-bit value assigned to a vendor.

D14F7x04 Status/Command

Bits	Description
31	**DetectedParityError: Detected Parity Error**. Read; Updated-by-hardware; Write-1-to-clear. Reset: 0. 1=The FCH detects a parity error.
30	**SignaledSystemError: Signaled System Error**. Read; Updated-by-hardware; Write-1-to-clear. Reset: 0. SERR# status. 1=The FCH detects a PCI address parity error.
29	**ReceivedMasterAbort: Received Master Abort**. Read; Updated-by-hardware; Write-1-to-clear. Reset: 0. 1=The FCH acts as a PCI master and aborts a PCI bus memory cycle.
28	**ReceivedTargetAbort: Received Target Abort**. Read; Updated-by-hardware; Write-1-to-clear. Reset: 0. 1=FCH received Target Abort from a PCI target.
27	**SignaledTargetAbort: Signaled Target Abort**. Read; Updated-by-hardware; Write-1-to-clear. Reset: 0. 1=The FCH signaled Target Abort while acting as Bus Master.
26:25	**DeviceSelectTiming: Device Select Timing**. Value: 01b. 01b=DEVSEL# Medium timing when performing a positive decode.
24	**MasterDataParityError: Master Data Parity Error**. Read; Updated-by-hardware; Write-1-to-clear. Reset: 0. Data Parity reported. 1=The FCH detects PERR# asserted while acting as PCI master.
23:21	Reserved.
20	**CapabilitiesList: Capabilities List**. Value: 1.
19:10	Reserved.
9	**FastBack2BackEnable: Fast Back-to-Back Enable**. Value: 0.
8	**SERREnable: SERR# Enable**. Read-write. Reset: 0. 1=FCH asserts SERR# when it detects as address parity error.
7	**SteppingControl: Stepping Control**. Value: 0. 1=Wait Cycle enable.
6	**ParityErrorResponse: Parity Error Response**. Read-write. Reset: 0. PERR# (Response) detection enable bit. 1=The FCH asserts PERR# when it is the agent receiving data and it detects a parity error. 0=PERR# is not asserted.
5	**VGAPaletteSnoop: VGA Palette Snoop**. Value: 0. 1=VGA Palette Snoop Enable. 0=Disable.
4	**MemoryWriteandInvalidateEnable: Memory Write and Invalidate Enable**. Value: 0. 1=Memory write and invalidate enable.
3	**SpecialCycles: Special Cycles**. Value: 0. 1=Special cycle recognition enable.

2	**BusMaster: Bus Master**. Read-write. Reset: 0. 1=Bus master function enable.
1	**MemorySpace: Memory Space**. Read-write. Reset: 0. 1=Enable memory access. 0=Disable.
0	**IOSpace: IO Space**. Value: 0. 1=IO access enable.

D14F7x08 Revision ID/Class Code

Bits	Description
31:24	**ClassCode: Class Code**. Read-only. Reset: 8h. Basic Class.
23:16	**SubClass: Sub Class**. Read-only. Reset: 5h. For SD host controller.
15:8	**InterfaceCode: Interface Code**. Value: 1. Bits Definition 00h Standard host not supporting DMA 01h Standard host supporting DMA 02h Vendor unique SD host controller FFh-03hReserved
7:0	**RevisionID: Revision ID**. Value: 01h.

D14F7x0C Cache Line Size

Bits	Description
31:24	**Bist**. Read-only. Reset: 0. No BIST modes.
23:16	**HeaderType: Header Type**. Read-only. Reset: 80h. Specifies the type of the predefined header in the configuration space.
15:8	**LatencyTimer: Latency Timer**. Read-only. Reset: 0. Specifies the value of the Latency Timer in units of PCICLKs.
7:0	**CacheLineSize: Cache Line Size**. Read-only. Reset: 0.

D14F7x10 Base Address Reg 0

Bits	Description
31:8	**BaseAddress0: Base Address 0**. Read-write. Reset: 0h. Specifies base address 0[31:8].
7:4	Reserved.
3	**Prefetchable**. Value: 0. 0=Non-prefetchable.
2:1	**Type**. Value: 10b. 00b=32-bit base address. 10b=64-bit base address.
0	**SpaceIndicator: Space Indicator**. Value: 0. 0=Memory-mapped base address.

D14F7x14 Upper Base Address Reg 0

Bits	Description
31:0	**BaseAddress0Upper: Base Address 0 Upper**. Read-write. Reset: 0. Specifies base address 0[63:32].

D14F7x2C Subsystem ID and Subsystem Vendor ID

Write-once. This 4-byte register is a write-once & read-only afterward register. The BIOS writes this register once (all 4 bytes at once) and software reads its value (when needed).

Bits	Description
31:16	**SubsystemID: Subsystem ID**. Reset: 7806h. Subsystem ID.
15:0	**SubsystemVendorID: Subsystem Vendor ID**. Reset: 1022h. Subsystem Vendor ID.

D14F7x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapabilitiesPointer: Capabilities Pointer**.Read-Only. Reset: 0.

D14F7x3C Interrupt Line

Bits	Description
31:16	Reserved.
15:8	**InterruptPin: Interrupt Pin**. Read-only. Reset: 1. Interrupt pin.
7:0	**InterruptLine: Interrupt Line**. Read-write. Reset: 0. Interrupt line.

D14F7x40 Slot Information

Bits	Description
31:7	Reserved.
6:4	**NumberOfSlots: Number of Slots**. Read-write. Reset: 0. Number of slots supported. These bits indicate the number of slots the Host Controller supports. In the case of a single function, a maximum of 6 slots can be assigned. Bits Definition Bits Definition 000b 1 slot 100b 5 slots 001b 2 slots 101b 6 slots 010b 3 slots 110b Reserved 011b 4 slots 111b Reserved

3	Reserved.
2:0	**FirstBaseAddressRegisterNumber: First Base Address Register Number**. Read-write. Reset: 0. Up to 6 base addresses can be specified in single configuration. These bits indicate the first base address register number assigned to SD Host Controller register set. In the case of single function and multiple register sets, contiguous base addresses are used. D14F7x40[NumberOfSlots] specifies the number of base addresses.

Bits	Definition	Bits	Definition
000b	Base Address 10h (BAR0)	100b	Base Address 20h (BAR4)
001b	Base Address 14h (BAR1)	101b	Base Address 24h (BAR5)
010b	Base Address 18h (BAR2)	110b	Reserved
011b	Base Address 1Ch (BAR3)	111b	Reserved

D14F7x80 SD PCI MSI Capability Header

Bits	Description
31:24	Reserved.
23	**ExtendAddrEn**. Value: 1. 64-bit Address Capable.
22:20	**MulMsgEn**. Read-write. Reset: 0. Multiple Message Enable.
19:17	**MulMsgCap**. Read-only. Reset: 0. Multiple Message Capable.
16	**MsiEnable**. Read-write. Reset: 0. MSI Enable.
15:8	**CapNxtPtr**. Value: 0. Reset: 0.
7:0	**CapId**. Read-only. Reset: 5h. A value of 05h indicates MSI.

D14F7x84 SD PCI MSI Address

Bits	Description
31:2	**MsgAddr**. Read-write. Reset: 0. Specifies Message Address[31:2].
1:0	Reserved.

D14F7x88 SD PCI MSI Upper Address

Bits	Description
31:0	**MsgUpperAddr**. Read-write. Reset: 0. Specifies Message Upper Address[63:32].

D14F7x8C SD PCI MSI Data

Bits	Description
31:16	Reserved.
15:0	**MsgData**. Read-write. Reset: 0. Message Data.

D14F7x90 Power Management Capability Header

Bits	Description
31:27	**PMESupport: PME Support**. Read-only. Reset: 0. This 5-bit field indicates the power states in which the function may assert PME#. A value of 0 for any bit indicates that the function is not capable of asserting the PME# signal while in that power state. Bit Definition [0] PME# can be asserted from D0 [1] PME# can be asserted from D1 [2] PME# can be asserted from D2 [3] PME# can be asserted from D3hot [4] PME# can be asserted from D3cold
26	**D2Support: D2 Support**. Read-only. Reset: 0. 1=This function supports the D2 Power Management State.
25	**D1Support: D1 Support**. Read-only. Reset: 0. 1=This function supports the D1 Power Management State.
24:22	**AuxCurrent: Aux Current**. Read-only. Reset: 0. This 3-bit field reports the 3.3V auxiliary current requirements for the PCI function.
21	**Dsi**. Read-only. Reset: 0. The Device Specific Initialization required. 1=Indicates special initialization of this function is required (beyond the standard PCI configuration header) before the generic class device driver is able to use it.
20	Reserved.
19	**PmeClock: PME Clock**. Value: 0. PME Clock. 0=Indicates that no PCI clock is required for the function to generate PME#. 1=Indicates that the function relies on the presence of the PCI clock for PME# operation.
18:16	**Version**. Value: 011b. 011b=Indicates that this function complies with revision 1.2 of the PCI Power Management Interface Specification.
15:8	**CAPNextPointer: CAP Next Pointer**. Read-only. Reset: 0. No other capability.
7:0	**CAPID: CAP ID**. Read-only. Reset: 1. Power management.

D14F7x94 Power Management Control and Status Register

Bits	Description
31:24	**Data**. Read-only. Reset: 0. Reports the state dependent data requested by the DataSelect field. The value of this register is scaled by the value reported by the DataScale field.
23	**BPCCEn: Bus Power/Clock Control Enable**. Read-only. Reset: 0. Bus Power/Clock Control Enable.
22	**B2B3: B2 B3#**. Read-only. Reset: 0. B2/B3 support for D3hot.
21:15	Reserved.
14:13	**DataScale: Data Scale**. Read-only. Reset: 0. Indicates the scaling factor to be used when interpreting the value of the Data register. The value and meaning of this field varies depending on which data value has been selected by the DataSelect field.
12:9	**DataSelect: Data Select**. Read-Write. Reset: 0. Selects which data is to be reported through the Data register and DataScale field.

8	**PMEEn: PME En**. Read-Write. Reset: 0. 1=Enable the function to assert PME#. 0=PME# assertion is disabled.
7:4	Reserved.
3	**NoSoftReset: No Soft Reset**. Read-only. Reset: 0. 1=Indicates that devices transitioning from D3hot to D0 because of PowerState commands do not perform an internal reset.
2	Reserved.
1:0	**Power**. Read-only. Reset: 0. Determines the current power state of a function and to set the function into a new power state. Bits Definition Bits Definition 00b D0 10b D2 01b D1 11b D3hot If software attempts to write an unsupported, optional state to this field, the write operation must complete normally on the bus; however, the data is discarded and no state change occurs.

D14F7xB8 SD Auto Pattern

Bits	Description
31:28	**SdDrvStr33Sn**. Read-write. Reset: 0. Specifies the SN driver strength value for 3.3V.
27:24	**SdDrvStr33Sp**. Read-write. Reset: 0. Specifies the SP driver strength value for 3.3V.
23:17	Reserved.
16	**ManualTune**. Read-write. Reset: 0. 1=Manual tune enable.
15:12	Reserved.
11	**AutoPatternEn**. Read-write. Reset: 0. 1=AUTO pattern enable.
10:5	Reserved.
4:0	**AutoTuneSel**. Read-write. Reset: 0h. RX clock select in AUTO PATTERN mode.

3.26.5.2 SD Host Controller Configuration Registers (SDHC)

The registers are accessed directly mapped by D14F7x10 [Base Address Reg 0] and D14F7x14 [Upper Base Address Reg 0].

SDHC0x00 SDHC System Address / Argument 2

Bits	Description
31:16	**SysAddr1**. Read-write. Reset: 0. System Address upper bits. Updating this register clears DMA_WAIT. It indicates system memory address for DMA. When DMA transfer detects the DMA Buffer Boundary specified by the Host DMA Buffer Boundary in the Block Size register, SD controller asserts DMA_WAIT. Also SD controller generates DMA interrupt at this time when corresponding bits in the Normal Interrupt Status Enable register (SDHC0x34) and Normal Interrupt Signal Enable register (SDHC0x38) are set. While ADMA is enabled, this register is not be used.
15:0	**SysAddr0**. Read-write. Reset: 0. System Address lower bits. Updating this register clears DMA_WAIT.

SDHC0x04 SDHC Block CS

Block Size and Block Count.

Bits	Description
31:16	**BlkCnt**. Read-write. Reset: 0. Block Count. It indicates block count of multiple data transfer. It is enabled when SDHC0x0C[BlkCntEn] == 1. It is decremented after each block data transmission. During infinite data transmission, setting this bit performs no function.
15	Reserved.
14:12	**DmaBufBndry**. Read-write. Reset: 0. Host DMA Buffer Boundary. Indicates the contiguous buffer size in the system memory. When the boundary is reached, DMA interrupt will be generated. Bits Definition Bits Definition 000b 4K bytes 100b 64K bytes 001b 8K bytes 101b 128K bytes 010b 16K bytes 110b 256K bytes 011b 32K bytes 111b 512K bytes
11:0	**BlkSize**. Read-write. Reset: 0. Transfer Data Length (max. block size is 2K bytes).

SDHC0x08 SDHC Command Argument

Bits	Description
31:16	**ARGUMENT1**. Read-write. Reset: 0. Upper bits. Command Argument. Command arguments specified as bits [39:8] of the command format.
15:0	**ARGUMENT0**. Read-write. Reset: 0. Lower bits.

SDHC0x0C SDHC Command/Transfer Mode

Bits	Description
31:30	Reserved.
29:24	**CmdIdx**. Read-write. Reset: 0. Command Index.
23:22	**CmdType**. Read-write. Reset: 0. Command Type. Bits Definition 00b Normal. 01b Suspend CMD52 for writing BR in CCCR. 10b Resume CMD52 for writing Function Sel in CCCR. 11b Abort CMD12 (SD Memory) or Abort CMD52 (SDIO).
21	**DataPrsnt**. Read-write. Reset: 0. Data Present Select. Indicates data is present and is transferred on the DAT line. When command is issued with this bit enabled, internal buffer is cleared. 0=No data. 1=Data.
20	**CmdIdxChkEn**. Read-write. Reset: 0. 1=Command Index Check Enable. 0=Disable.
19	**CrcChkEn**. Read-write. Reset: 0. 1=Command CRC Check Enable. 0=Disable.
18	Reserved.

17:16	**RespType**. Read-write. Reset: 0. Response Type Select. Bits Definition 00b No Response. 01b Response length is 136 bits. 10b Response length is 48 bits without busy. 11b Response length is 48 bits with busy.
15:6	Reserved.
5	**MultiBlk**. Read-write. Reset: 0. 1=Multiple/Single Block Select. 0=Single Block.
4	**DataDir**. Read-write. Reset: 0. Data Transfer Direction. 0=Write. 1=Read.
3:2	**AutoCmdEn**. Read-write. Reset: 0. Bits Definition 00b Auto Command Disabled 01b Auto CMD12 Enable 10b Auto CMD23 Enable 11b Reserved
1	**BlkCntEn**. Read-write. Reset: 0. 1=Block Count Enable.
0	**DmaEn**. Read-write. Reset: 0. 1=DMA Enable. 0=Disable.

SDHC0x10 SDHC_RESP1_0

Bits	Description
31:16	**RESPONSE1**. Read-only. Reset: 0. R39-24 of response is saved in this field.
15:0	**RESPONSE0**. Read-only. Reset: 0. R23-8 of response is saved in this field.

SDHC0x14 SDHC_RESP3_2

Bits	Description
31:16	**RESPONSE3**. Read-only. Reset: 0. R71-56 of response is saved in this field.
15:0	**RESPONSE2**. Read-only. Reset: 0. R55-40 of response is saved in this field.

SDHC0x18 SDHC_RESP5_4

Bits	Description
31:16	**RESPONSE5**. Read-only. Reset: 0. R103-88 of response is saved in this field.
15:0	**RESPONSE4**. Read-only. Reset: 0. R87-72 of response is saved in this field.

SDHC0x1C SDHC_RESP7_6

Bits	Description
31:16	**RESPONSE7**. Read-only. Reset: 0. R127-120 of response or R39-24 of Auto CMD12 response is saved in this field.
15:0	**RESPONSE6**. Read-only. Reset: 0. R119-104 of response or R23-8 of Auto CMD12 response is saved in this field.

SDHC0x20 SDHC_BUFFER

Bits	Description
31:16	**BuffData1**. Read-write. Reset: 0. Upper bits. Data Buffer. Data will be accessed through this register. Data which exceeds the size designated by the Block Size register is not be written in the data buffer.
15:0	**BuffData0**. Read-write. Reset: 0. Lower bits.

SDHC0x24 SDHC_PRSNT_STATE

Bits	Description
31:25	Reserved.
24	**CmdLevel**. Read-only. Reset: 0. CMD Line Signal Level. Reflects signal level of CMD line.
23:20	**DatLevel**. Read-only. Reset: 0. DAT Line Signal Level. Reflects signal level of DAT line.
19	**WpLevel**. Read-only. Reset: 0. Write Protect Switch Level. 0=Write protected. 1=Write enable.
18	**CdLevel**. Read-only. Reset: 0. Card Detect Pin Level. 0=No card present. 1=Card present.
17	**CardStable**. Read-only. Reset: 0. Card State Stable. Indicates Card Detect signal level is stable. 0=Not stable (debouncing or resetting). 1=Card stable.
16	**CardIns**. Read-only. Reset: 0. Card Inserted. 0=No card inserted or debouncing state or resetting. 1=Card inserted.
15:12	Reserved.
11	**BufRdEn**. Read-only. Reset: 0. Buffer Read Enable. Indicates buffer is ready for reading. 0=Read disable. 1=Read enable.
10	**BufWrEn**. Read-only. Reset: 0. Buffer Write Enable. Indicates buffer is ready for writing. 0=Write disable. 1=Write enable.
9	**RdTxActive**. Read-only. Reset: 0. Read Transfer Active. Indicates occurrence of read data transfer. 0=No data transferring. 1=Read data transferring.
8	**WrTxActive**. Read-only. Reset: 0. Write Transfer Active. Indicates occurrence of write data transfer. 0=No data transferring. 1=Write data transferring.
7:3	Reserved.
2	**DatLineActive**. Read-only. Reset: 0. DAT Line Active. Indicates DAT line on SD Bus is active. 0=DAT line inactive. 1=DAT line active.

1	**CmdInhibDat**. Read-only. Reset: 0. Command Inhibit (DAT). Indicates that commands which use also the DAT line can be issued. 0=Can issue commands which use DAT line. 1=Cannot issue any commands which use DAT line.
0	**CmdInhibCmd**. Read-only. Reset: 0. Command Inhibit (CMD). Indicates that commands which use only the CMD line can be issued. 0=Can issue commands which use CMD line. 1=Cannot issue any commands.

SDHC0x28 SDHC_CTRL1

Bits	Description
31:27	Reserved.
26	**SdRemWakeEn**. Read-write. Reset: 0. 1=SD Card Removal Wake up. 0=Disable.
25	**SdInsWakeEn**. Read-write. Reset: 0. 1=SD Card Insertion Wake up. 0=Disable.
24	**SdIntWakeEn**. Read-write. Reset: 0. 1=SD Card Interrupt Wake up. 0=Disable.
23:20	Reserved.
19	**BgIntEn**. Read-write. Reset: 0. Interrupt at Block Gap. 1=Enable interrupt detection during 4-bit block transmission. 0=Disable.
18	**ReadWaitEn**. Read-write. Reset: 0. Read Wait Control. 1=Indicates Read Wait will be inserted when needed. 0=Disable.
17	**ContReq**. Read-write. Reset: 0. Continue Request. Writing 1 to this bit triggers restart of halted data transaction with current register setting. Once this bit is 1, the internal buffer will be cleared and data transfer sequence will be restarted. 1=Restart.
16	**BgStopReq**. Read-write. Reset: 0. Stop at Block Gap Request. Writing 1 to this bit triggers halting of current data transfer after next block gap. To use this request, the Read Wait function is necessary in read transaction. Even if SDHC0x0C[AutoCmdEn]==01b, Auto CMD12 is not issued in case this bit is set to 1. This bit is cleared by not only writing 0 to this bit, but also issuing abort commands. 0=Transfer. 1=Stop.
15:12	Reserved.
11:9	**SdBusVoltage**. Read-write. Reset: 0. SD Bus Voltage. Bits / Definition / Bits / Definition 100b-000b Reserved 110b 3.0 Volts 101b 1.8 Volts 111b 3.3 Volts
8	**SdBusEn**. Read-write. Reset: 0. SD Bus Power. Automatically cleared when card is removed. 0=Off. 1=On.
7	**CdTestEn**. Read-write. Reset: 0. Card Detect Signal Selection. 0=IO pin. 1=SD_TEST_LEVEL.
6	**CdTestLevel**. Read-write. Reset: 0. Card Detect Test Level. 0=Card removed. 1=Card inserted.
5	**MmcWidth**. Read-write. Reset: 0. Extended Data Transfer Width. 0=Use width set by DAT_TX_WIDTH. 1=Force 8-bit.
4:3	**DmaSelect**. Read-write. Reset: 0. DMA Select. Valid only when DMA is enabled. Bits / Definition / Bits / Definition 00b No DMA or SDMA selected 10b 32-bit ADMA2 01b 32-bit ADMA 11b 64-bit ADMA2

2	**HighSpeedEn**. Read-write. Reset: 0. High Speed Enable. When disabled, SD controller outputs commands and data on the falling edge of the SD clock. (Up to 25 MHz SD clock can be supported.) When enabled, SD controller outputs commands and data on the rising edge of the SD clock. (Up to 50 MHz SD clock can be supported.) 0=Normal speed. 1=High speed.
1	**DatTxWidth**. Read-write. Reset: 0. Data Transfer Width. 0=1-bit. 1=4-bit.
0	**LedCtrl**. Read-write. Reset: 0. LED control. 1=Drives the LED_ON output. 0=Off.

SDHC0x2C SDHC_CTRL2

Bits	Description
31:27	Reserved.
26	**SoftRstDat**. Read-write. Reset: 0. Software Reset for DAT Line. The following registers will be cleared: SDHC0x20[BuffData1,BuffData0], SDHC0x24[BufRdEn, BufWrEn, RdTxActive, WrTxActive, DatLineActive, CmdInhibDat], SDHC0x28[ContReq, BgStopReq], SDHC0x30[BufRdRdy, BufWrRdy, BlockGapEvt, DatDone].
25	**SoftRstCmd**. Read-write. Reset: 0. Software Reset for CMD Line. The following registers will be cleared: SDHC0x24[CmdInhibCmd], SDHC0x30[CmdDone].
24	**SoftRstAll**. Read-write. Reset: 0. Software Reset for All. The following registers will not be cleared: CMD Line Signal Level- DAT[3:0] Line Signal Level, Write Protect Switch Pin Level, Card Detect Pin Level, Card State Stable, Card Inserted (see SDHC0x24) and all bits in the Capabilities Register (SDHC0x28).
23:20	Reserved.
19:16	**DataToCnt**. Read-write. Reset: 0. Data Timeout Counter Value. By using this counter value, DAT line timeouts are detected. Bits: Eh-0h — Definition: 2^(<DataToCnt>+13) Bits: Fh — Definition: Reserved.
15:8	**SdclkDiv[7:0]**. Read-write. Reset: 0. SDCLK Frequency Select. If multiple bits are set, the most significant bit will be selected. Host Controller Version 1.00 and 2.00: 0000_0000b: Divide by 1. 0000_0001b: Divide by 2. 0000_001Xb: Divide by 4. 0000_01XXb: Divide by 8. 0000_1XXXb: Divide by 16. 0001_XXXXb: Divide by 32. 001X_XXXXb: Divide by 64. 01XX_XXXXb: Divide by 128. 1XXX_XXXXb: Divide by 256. Host Controller Version 3.00, SdclkDiv[9:0] = {SdcklDiv[9:8], SdclkDiv[7:0]}: 3FFh-001h: Divide by 1/(2*SdclkDiv[9:0]) 000h: Divide by 1
7:6	**SdclkDiv[9:8]**. Read-write. Reset: 0. See SdclkDiv[7:0].
5	**ClkGenSel: Clock Generator Select**. Read-write. Reset: 0. 1=Programmable Clock Mode. 0=Divided Clock Mode.
4:3	Reserved.
2	**SdclkEn**. Read-write. Reset: 0. SD Clock Enable. SDCLK Frequency Select can be changed when this bit is 0. When card is removed, this bit is cleared to 0 automatically. 0=Disable. 1=Enable.

1	**SysclkStable**. Read-only. Reset: 0. Internal Clock Stable. 0=Unstable. 1=Stable.
0	**SysclkEn**. Read-write. Reset: 0. 1=Internal Clock Enable. 0=Disable.

SDHC0x30 SDHC_INT_STATUS

Bits	Description
31:30	Reserved.
29	**CeAtaErr**. Read; Write-1-to-clear. Reset: 0. CE-ATA Error.
28	**SdmaErr**. Read; Write-1-to-clear. Reset: 0. SDMA Error.
27	Reserved.
26	**TuningError**. Read; Write-1-to-clear. Reset: 0. 1=An unrecoverable tuning error is detected in a during the tuning procedure.
25	**AdmaErr**. Read; Write-1-to-clear. Reset: 0. ADMA Error.
24	**Acmd12Err**. Read; Write-1-to-clear. Reset: 0. Auto CMD12 Error. Logical OR of Auto CMD12 Error Status Register.
23	Reserved.
22	**DatEndErr**. Read; Write-1-to-clear. Reset: 0. Data CRC Error.
21	**DatCrcErr**. Read; Write-1-to-clear. Reset: 0. Data CRC Error.
20	**DatToErr**. Read; Write-1-to-clear. Reset: 0. Data Timeout Error.
19	**CmdIdxErr**. Read; Write-1-to-clear. Reset: 0. Command Index Error. Mismatch of Command Index and index of response.
18	**CmdEndErr**. Read; Write-1-to-clear. Reset: 0. Command End Bit Error.
17	**CmdCrcErr**. Read; Write-1-to-clear. Reset: 0. Command CRC Error. If both CmdToErr and CmdCrcErr are set, this indicates Command Conflict Error.
16	**CmdToErr**. Read; Write-1-to-clear. Reset: 0. Command Timeout Error. Response not returned within 128 SDCLK cycles.
15	**Error**. Read; Write-1-to-clear. Reset: 0. Error Interrupt.
14:13	Reserved.
12	**ReTuningEvent.** Read-only. Reset: 0. 1=Re-tuning should be performed.
11:9	Reserved.
8	**Sdio**. Read; Write-1-to-clear. Reset: 0. SDIO Card Interrupt. Writing 1 to this register does not clear this bit. To clear this bit, interrupt factor of SDIO cards should be cleared. The value of this bit is latched internally as long as the Card Interrupt bit in the Normal Interrupt Status Enable register (SDHC0x34[SdioMask]) == 1.
7	**CardRem**. Read; Write-1-to-clear. Reset: 0. Card Removal.
6	**CardIns**. Read; Write-1-to-clear. Reset: 0. Card Insertion.
5	**BufRdRdy**. Read; Write-1-to-clear. Reset: 0. Buffer Read Ready. In the case where SDHC0x0C[AutoCmdEn] == 01b and last block has been transferred, Auto CMD12 will be issued prior to this bit being set to 1. Clearing this bit should be done before buffer reading, because SD controller has dual buffer and the next Buffer Read Ready interrupt may occur immediately.

4	**BufWrRdy**. Read; Write-1-to-clear. Reset: 0. Buffer Write Ready. Clearing this bit should be done before buffer writing, because the SD controller has dual buffer and the next Buffer Write Ready interrupt may occur immediately.
3	**DmaEvt**. Read; Write-1-to-clear. Reset: 0. DMA Interrupt. It is set when internal counter reaches the value designated by Host DMA Buffer Boundary (SDHC0x04[DmaBufBndry]). It should be cleared by Host Driver after System Address Register (SDHC0x00) is updated.
2	**BlockGapEvt**. Read; Write-1-to-clear. Reset: 0. Block Gap Event. It indicates the timing of next block gap, which was requested by SDHC0x28[BgStopReq]. In case of write transaction, this interrupt will be generated before busy completion.
1	**DatDone**. Read; Write-1-to-clear. Reset: 0. Data Transfer Complete. Indicates the timing for completion of data transaction, which includes the completion at the block gap by SDHC0x28[BgStopReq]. When some errors are detected during data transaction, this bit will not be set. In the case where SDHC0x0C[AutoCmdEn] == 01b, Auto CMD12 will be issued prior to this bit being set to 1.
0	**CmdDone: cmdDone (r)**. Read-write. Reset: 0. Command Complete. The end bit of the command response is received. In the case of commands with no response, the end of the command.

SDHC0x34 SDHC_INT_MASK

Bits	Description
31:30	Reserved.
29	**CeAtaErrMask**. Read-write. Reset: 0. CE-ATA Error. 0=Masked. 1=Enable.
28	**SdmaErrMask**. Read-write. Reset: 0. SDMA Error. 0=Masked. 1=Enable.
27	Reserved.
26	**TuningErrorStatusEnable.** Read-write. Reset: 0. 1=Tuning Event Status Enable. 0=Masked.
25	**AdmaErrMask**. Read-write. Reset: 0. ADMA Error. 0=Masked. 1=Enable.
24	**Acmd12ErrMask**. Read-write. Reset: 0. Auto CMD12 Error. 0=Masked. 1=Enable.
23	**CurLimErrMask**. Read-write. Reset: 0. Current Limit Error. 0=Masked. 1=Enable.
22	**DatEndErrMask**. Read-write. Reset: 0. Data End Bit Error. 0=Masked. 1=Enable.
21	**DatCrcErrMask**. Read-write. Reset: 0. Data CRC Error. 0=Masked. 1=Enable.
20	**DatToErrMask**. Read-write. Reset: 0. Data Timeout Error. 0=Masked. 1=Enable.
19	**CmdIdxErrMask**. Read-write. Reset: 0. Command Index Error. 0=Masked. 1=Enable.
18	**CmdEndErrMask**. Read-write. Reset: 0. Command End Bit Error. 0=Masked. 1=Enable.
17	**CmdCrcErrMask**. Read-write. Reset: 0. Command CRC Error. 0=Masked. 1=Enable.
16	**CmdToErrMask**. Read-write. Reset: 0. Command Timeout Error. 0=Masked. 1=Enable.
15:13	Reserved.
12	**ReTuningEventStatusEn**. Read-write. Reset: 0. 1=Re-Tuning Event Status Enable. 0=Masked.
11:9	Reserved.
8	**SdioMask**. Read-write. Reset: 0. Card Interrupt. 0=Masked. 1=Enable.
7	**CardRemMask**. Read-write. Reset: 0. Card Removal. 0=Masked. 1=Enable.
6	**CardInsMask**. Read-write. Reset: 0. Card Insertion. 0=Masked. 1=Enable.
5	**BufRdRdyMask**. Read-write. Reset: 0. Buffer Read Ready. 0=Masked. 1=Enable.

4	**BufWrRdyMask**. Read-write. Reset: 0. Buffer Write Ready. 0=Masked. 1=Enable.
3	**DmaEvtMask**. Read-write. Reset: 0. DMA Interrupt. 0=Masked. 1=Enable.
2	**BlockGapEvtMask**. Read-write. Reset: 0. Block Gap Event. 0=Masked. 1=Enable.
1	**DatDoneMask**. Read-write. Reset: 0. Transfer Complete. 0=Masked. 1=Enable.
0	**CmdDoneMask**. Read-write. Reset: 0. Command Complete. 0=Masked. 1=Enable.

SDHC0x38 SDHC_SIG_MASK

Bits	Description
31:30	Reserved.
29	**CeAtaErrEn**. Read-write. Reset: 0. CE-ATA Error. 0=Masked. 1=Enable.
28	**SdmaErrEn**. Read-write. Reset: 0. SDMA Error. 0=Masked. 1=Enable.
27	Reserved.
26	**TuningErrorStatusEnable.** Read-write. Reset: 0. 1=Tuning Event Status Enable. 0=Masked.
25	**AdmaErrEn**. Read-write. Reset: 0. ADMA Error. 0=Masked. 1=Enable.
24	**Acmd12ErrEn**. Read-write. Reset: 0. Auto CMD12 Error. 0=Masked. 1=Enable.
23	**CurLimErrEn**. Read-write. Reset: 0. Current Limit Error. 0=Masked. 1=Enable.
22	**DatEndErrEn**. Read-write. Reset: 0. Data End Bit Error. 0=Masked. 1=Enable.
21	**DatCrcErrEn**. Read-write. Reset: 0. Data CRC Error. 0=Masked. 1=Enable.
20	**DatToErrEn**. Read-write. Reset: 0. Data Timeout Error. 0=Masked. 1=Enable.
19	**CmdIdxErrEn**. Read-write. Reset: 0. Command Index Error. 0=Masked. 1=Enable.
18	**CmdEndErrEn**. Read-write. Reset: 0. Command End Bit Error. 0=Masked. 1=Enable.
17	**CmdCrcErrEn**. Read-write. Reset: 0. Command CRC Error. 0=Masked. 1=Enable.
16	**CmdToErrEn**. Read-write. Reset: 0. Command Timeout Error. 0=Masked. 1=Enable.
15:13	Reserved.
12	**ReTuningEventStatusEn**. Read-write. Reset: 0. 1=Re-Tuning Event Status Enable. 0=Masked.
11:9	Reserved.
8	**SdioEn**. Read-write. Reset: 0. Card Interrupt. 0=Masked. 1=Enable.
7	**CardRemEn**. Read-write. Reset: 0. Card Removal. 0=Masked. 1=Enable.
6	**CardInsEn**. Read-write. Reset: 0. Card Insertion. 0=Masked. 1=Enable.
5	**BufRdRdyEn**. Read-write. Reset: 0. Buffer Read Ready. 0=Masked. 1=Enable.
4	**BufWrRdyEn**. Read-write. Reset: 0. Buffer Write Ready. 0=Masked. 1=Enable.
3	**DmaEvtEn**. Read-write. Reset: 0. DMA Interrupt. 0=Masked. 1=Enable.
2	**BlockGapEvtEn**. Read-write. Reset: 0. Block Gap Event. 0=Masked. 1=Enable.
1	**DatDoneEn**. Read-write. Reset: 0. Transfer Complete. 0=Masked. 1=Enable.
0	**CmdDoneEn**. Read-write. Reset: 0. Command Complete. 0=Masked. 1=Enable.

SDHC0x3C SDHC ACMD12 Error/Host Control 2

Bits	Description
31	**PresetValEn.** Read-write. Reset: 0. 1=Enable automatic selection of Preset Value. 0=SDCLK and driver strength are controlled by Host Driver.
30	**AsyncIntEn.** Read-write. Reset: 0. Enable Asynchronous Interrupts.
29:24	Reserved.
23	**SamplingClkSel.** Read-write. Reset: 0. 1=Tuned clock is used to sample data. 0=Fixed clock is used to sample data.
22	**ExecTuning.** Read-write; Cleared-when-done. Reset: 0. 1=Execute Tuning. 0=Not Tuned or Tuning Completed.
21:20	**DvrStrengthSel.** Read-write. Reset: 0. Bits Definition Bits Definition 00b Driver Type B is Selected (Default) 10b Driver Type C is Selected 01b Driver Type A is Selected 11b Driver Type D is Selected
19	**En_1_8VSignaling.** Read-write. Reset: 0. 1=1.8 Volt Signaling. 0=3.3 Volt Signaling.
18:16	**UHS_ModeSel.** Read-write. Reset: 0. Select UHS-I mode when (En_1_8VSignaling == 1). Bits Definition Bits Definition 000b SDR12 011b SDR104 001b SDR25 100b DDR50 010b SDR50 111b-101b Reserved
15:8	Reserved.
7	**CmdErr**. Read-only. Reset: 0. Command Not Issued By Auto CMD12 Error.
6:5	Reserved.
4	**IndexErr**. Read-only. Reset: 0. Auto CMD12 Index Error.
3	**EndErr**. Read-only. Reset: 0. Auto CMD12 End Bit Error.
2	**CrcErr**. Read-only. Reset: 0. Auto CMD12 CRC Error.
1	**ToErr**. Read-only. Reset: 0. Auto CMD12 Timeout Error.
0	**ExeErr**. Read-only. Reset: 0. Auto CMD12 Not Executed Error.

SDHC0x40 SDHC_CAPABILITY

Bits	Description
31:30	**SlotType**. Read-only. Reset: 0. Bits Definition Bits Definition 00b Removable Card Slot 10b Shared Bus Slot 01b Embedded Slot for One Device 11b Reserved.
29	**AsynIntSup: Asynchronous Interrupt Support**. Read-only. Reset: 1. 1=Asynchronous interrupt supported. 0=Asynchronous interrupt not supported.
28	**Support64BitSystemBus**. Read-only. Reset: 0. 1=Host Controller supports 64-bit address descriptor mode and 64-bit address system bus.
27	Reserved.

26	**Support18v**. Read-only. Reset: 1. 1=Voltage Support for 1.8V.
25	**Support30v**. Read-only. Reset: 0. 1=Voltage Support for 3.0V.
24	**Support33v**. Read-only. Reset: 1. 1=Voltage Support for 3.3V.
23	**SusResSupport**. Read-only. Reset: 1. 1=Suspend and Resume is supported.
22	**DmaSupport**. Read-only. Reset: 1. 1=DMA is supported.
21	**HiSpeedSupport**. Read-only. Reset: 1. 1=High Speed is supported.
20	**AdmaSupport**. Read-only. Reset: 1. 1=Advanced DMA is supported.
19	**Adma2Support**. Read-only. Reset: 1. 1=Advanced DMA2 is supported.
18	**Mmc8Support**. Read-only. Reset: 0. Extended Media Bus Support (MMC).
17:16	**MaxBlkLen**. Read-only. Reset: 2h. Specifies the maximum block size that the Host Driver can read and write to the buffer in the Host Controller. The buffer shall transfer this block size without wait cycles. Bits Definition Bits Definition 00b 512 bytes 10b 2048 bytes 01b 1024 bytes 11b Reserved
15:8	**BaseClkFreq**. Read-only. Reset: 32h. Specifies the base clock frequency for SD Clock. Bits Definition 00h Reserved FFh-01h <BaseClkFreq> MHz
7	**TmoClkUnit: Timeout Clock Unit**. Read-only. Reset: 1. Specifies the unit of TmoClkFreq. 0=Unit is KHz. 1=Unit is MHz.
6	Reserved.
5:0	**TmoClkFreq: Timeout Clock Frequency**. Read-only. Reset: 32h. Specifies the base clock frequency used to detect Data Timeout Error. TmoClkUnit defines the unit of this field's value.

SDHC0x44 SDHC_CAPABILITY 2

Bits	Description
31:24	Reserved.
23:16	**ClockMultiplier**. Read-only. Reset: 3. Bits Definition 00h ClockMultiplier not supported. FFh-01h <ClockMultiplier+1> MHz
15:14	**ReTuningModes**. Read-only. Reset: 0. Specifies re-tuning method and limits the maximum data length. Bits Definition 00b Mode 1, Timer, 4 MBMax Data length. 01b Mode 2, Timer and Re-Tuning Request, 4 MBMax Data length. 10b Mode 3, Auto Re-Tuning (for transfer)Timer and Re-Tuning Request, any Data Length. 11b Reserved.
13	**UseTuningForSDR50**. Read-only. Reset: 0. 1=SDR50 requires tuning.
12	Reserved.

11:8	**TimerCountForReTuning**. Read-only. Reset: 5. Specifies value of the Re-Tuning Timer for Re-Tuning Mode 1 to 3. Bits Definition 0h Re-tuning timer disabled. 1h 1 Second Bh-2h <2^(TimerCountForReTuning-1)> Seconds Eh-Ch Reserved Fh Get information from other source.
7	Reserved.
6	**DriverTypeDSupport**. Read-only. Reset: 0. 1=Support of Driver Type D for 1.8 Signaling.
5	**DriverTypeCSupport**. Read-only. Reset: 0. 1=Support of Driver Type C for 1.8 Signaling.
4	**DriverTypeASupport**. Read-only. Reset: 0. 1=Support of Driver Type A for 1.8 Signaling.
3	Reserved.
2	**DDR50Support**. Read-only. Reset: 0. 1=DDR50 is supported.
1:0	**SDRSupport**. Read-only. Reset: 3. Bit Definition [0] 1=SDR50 is supported. [1] 1=SDR104 is supported.

SDHC0x48 SDHC_CURR_CAPABILITY

Bits	Description
31:24	Reserved.
23:16	**MaxCurr18v**. Read-only. Reset: C8h. Max Current for 1.8V.
15:8	**MaxCurr30v**. Read-only. Reset: 0. Max Current for 3.0V.
7:0	**MaxCurr33v**. Read-only. Reset: 64h. Max Current for 3.3V.

SDHC0x50 SDHC_FORCE_EVT

Bits	Description
31:30	Reserved.
29	**CeAtaErrFrc**. Read-write. Reset: 0. Force CE-ATA Error.
28	**SdmaErrFrc**. Read-write. Reset: 0. Force SDMA Error.
27:26	Reserved.
25	**AdmaErrFrc**. Read-write. Reset: 0. Force ADMA Error.
24	**Acmd12ErrFrc**. Read-write. Reset: 0. Force Auto CMD12 Error.
23	**CurLimErrFrc**. Read-write. Reset: 0. Force Current Limit Error.
22	**DatEndErrFrc**. Read-write. Reset: 0. Force Data End Bit Error.
21	**DatCrcErrFrc**. Read-write. Reset: 0. Force Data CRC Error.
20	**DatToErrFrc**. Read-write. Reset: 0. Force Data Timeout Error.
19	**CmdIdxErrFrc**. Read-write. Reset: 0. Force Command Index Error.

18	**CmdEndErrFrc**. Read-write. Reset: 0. Force Command End Bit Error.
17	**CmdCrcErrFrc**. Read-write. Reset: 0. Force Command CRC Error.
16	**CmdToErrFrc**. Read-write. Reset: 0. Force Command Timeout Error.
15:8	Reserved.
7	**Acmd12CmdErrFrc**. Read-write. Reset: 0. Force Command Not Issued By Auto CMD12 Error.
6:5	Reserved.
4	**Acmd12IdxErrFrc**. Read-write. Reset: 0. Force Auto CMD12 Index Error.
3	**Acmd12EndErrFrc**. Read-write. Reset: 0. Force Auto CMD12 End Bit Error.
2	**Acmd12CrcErrFrc**. Read-write. Reset: 0. Force Auto CMD12 CRC Error.
1	**Acmd12ToErrFrc**. Read-write. Reset: 0. Force Auto CMD12 Timeout Error.
0	**Acmd12ExeErrFrc**. Read-write. Reset: 0. Force Auto CMD12 Not Executed Error.

SDHC0x54 SDHC_ADMA_ERR

Bits	Description
31:3	Reserved.
2	**AddrLenMismatch**. Read-only. Reset: 0. ADMA Address Length Mismatch Error. This error occurs in the following 2 cases: 1). While SDHC0x0C[BlkCntEn] == 1, the total data length specified by the descriptor table is different from that specified by SDHC0x04[BlkCnt] and SDHC0x04[BlkSize]. 2). Total data length cannot be divided by the block length.
1:0	**AdmaState**. Read-only. Reset: 0. ADMA State when error has occurred. Bits — Definition 00b — Stop DMA 01b — Fetch Descriptor. 10b — Change Address. 11b — Transfer Data.

SDHC0x58 SDHC_ADMA_SAD

Bits	Description
31:16	**AdmaSysAddr1**. Read-write. Reset: 0. Upper bits. ADMA System Address. Before ADMA data transfer, the descriptor address should be set by the Host Driver. This address needs to be set with 4-byte alignment, since the descriptor table has 32-bit (4 byte) information formatted.
15:0	**AdmaSysAddr0**. Read-write. Reset: 0. Lower bits.

SDHC0x6[C:0:step4] Preset Value

Table 276: Register Mapping for SDHC0x6[C:0:step4]

Register	Function
SDHC0x60	Initialization 3.3V or 1.8V/Default Speed 3.3V

Table 276: Register Mapping for SDHC0x6[C:0:step4] (Continued)

SDHC0x64	High Speed 3.3V/SDR12 1.8V Preset Values
SDHC0x68	SDR25 1.8V/SDR50 1.8V Preset Values
SDHC0x6C	SDR104 1.8V/DDR50 1.8V Preset Values

Table 277: Field Mapping for SDHC0x6[C:0:step4]

Register	Bits					
	31:30	26	25:16	15:14	10	9:0
SDHC0x60	DefaultSpeed	DefaultSpeed	DefaultSpeed	Initialization	Initialization	Initialization
SDHC0x64	SDR12	SDR12	SDR12	HighSpeed	HighSpeed	HighSpeed
SDHC0x68	SDR50	SDR50	SDR50	SDR25	SDR25	SDR25
SDHC0x6C	DDR50	DDR50	DDR50	SDR104	SDR104	SDR104

Bits	Description
31:30	**DvrStrengthSelPreset1**. See: DvrStrengthSelPreset0.
29:27	Reserved.
26	**ClkGenSelPreset1**. See: ClkGenSelPreset0.
25:16	**SdClkFreqSel1**. See: SdClkFreqSel0.
15:14	**DvrStrengthSelPreset0**. Read-only. Driver Strength is supported by 1.8V signaling bus speed modes. Does not apply for 3.3V signaling. See SDHC0x3C[DvrStrengthSel]. Bits Definition Bits Definition 00b Driver Type B is Selected 10b Driver Type C is Selected 01b Driver Type A is Selected 11b Driver Type D is Selected
13:11	Reserved.
10	**ClkGenSelPreset0**. Read-only. Clock Generator Select Value. See SDHC0x2C[ClkGenSel]. 1=Programmable Clock Generator. 0=Host Controller Ver2.00 Compatible Clock Generator.
9:0	**SdClkFreqSel0**. Read-only. A10-bit preset value. See SDHC0x2C[SdclkDiv[9:0]].

SDHC0xE0 Shared Bus Control Register

Bits	Description
31	Reserved.
30:24	**BackEndPwrCtrl: Back-End Power Control**. Read-write. Reset: 0. 0=Back-end power is off. 1=Back-end power is supplied. Bit Definition [0] Back-end Power Control for Device 1 [1] Back-end Power Control for Device 2 [2] Back-end Power Control for Device 3 [3] Back-end Power Control for Device 4 [4] Back-end Power Control for Device 5 [5] Back-end Power Control for Device 6 [6] Back-end Power Control for Device 7
23	Reserved.

22:20	**IntPinSel**. Read-write. Reset: 0. Selects interrupt pin inputs. Bits Definition Bits Definition 000b Interrupt is detected by Interrupt Cycle 010b INT_B is Enabled 001b INT_A is Enabled 100b INT_C is Enabled
19	Reserved.
18:16	**ClkPinSel**. Read-write. Reset: 0. Selects one of clock pin outputs. Bits Definition Bits Definition 000b Clock Pins are Disabled 100b CLK[4] is Selected 001b CLK[1] is Selected 101b CLK[5] is Selected 010b CLK[2] is Selected 110b CLK[6] is Selected 011b CLK[3] is Selected 111b CLK[7] is Selected
15	Reserved.
14:8	**BusWidthPreset**. Read-only. 0=4-bit bus width mode (SDHC0x28[DatTxWidth]==1). 1=8-bit bus width mode. Bit Definition [0] Bus Width Preset for Device 1 [1] Bus Width Preset for Device 2 [2] Bus Width Preset for Device 3 [3] Bus Width Preset for Device 4 [4] Bus Width Preset for Device 5 [5] Bus Width Preset for Device 6 [6] Bus Width Preset for Device 7
7:6	Reserved.
5:4	**NumIntInputPins.** Read-only. Indicates support of interrupt input pins INT_A#, INT_B# and INT_C# for shared bus system. Bits Definition Bits Definition 00b Interrupt Input Pin is Not Supported 10b INTA and INTB are Supported 01b INTA is Supported 11b INTA, INTB and INTC are Supported
3	Reserved.
2:0	**NumofClkPins**. Read-only. This field indicates support of clock pins to select one of devices for shared bus system. Bits Definition Bits Definition 000b Shared bus is not supported 100b 4 SDCLK pin is supported 001b 1 SDCLK pin is supported 101b 5 SDCLK pin is supported 010b 2 SDCLK pin is supported 110b 6 SDCLK pin is supported 011b 3 SDCLK pin is supported 111b 7 SDCLK pin is supported

SDHC0xFC SDHC_VER_SLOT

Bits	Description
31:24	**VendorVersion**. Read-only. Reset: 0. Vendor version.
23:16	**SpecVersion**. Read-only. Reset: 0. Specification version.

15:8	Reserved.
7:0	**SlotIntrpt**. Read-only. Reset: 0. Interrupt signal for each slot. The value of XSLT_INT7-0 inputs, which indicates the logical OR of Interrupt signal and Wakeup signal, are inverted and referred to by this register. In case of multiple slots, Interrupt signal and Wakeup signal should be logical OR'ed externally and should be inputted to each of XSLT_INT7-0.

3.26.6 SMBus Host Controller

3.26.6.1 Device 14h Function 0 (SMBus) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D14F0x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: Device Identifier**. Value: 790Bh. This 16-bit field is assigned by the device manufacturer and identifies the type of device.
15:0	**VendorID**. Read-only. Reset: 1022h. Vendor Identifier.

D14F0x04 Status/Command

Bits	Description
31	**DetectedParityError: Detected Parity Error**. Read; Write-1-to-clear. Reset: 0. This bit is set by the device whenever it detects a parity error, even if parity error handling is disabled.
30	**SignaledSystemError: Signaled System Error**. Read; Write-1-to-clear. Reset: 0. This bit is set by the device whenever the device asserts SERR#.
29	**ReceivedMasterAbort: Received Master Abort**. Read; Write-1-to-clear. Reset: 0. This bit is set by a slave device whenever it terminates its transaction with master abort.
28	**ReceivedTargetAbort: Received Target Abort**. Read; Write-1-to-clear. Reset: 0. This bit is set by a master device whenever its transaction is terminated with target abort.
27	**SignaledTargetAbort: Signaled Target Abort**. Read; Write-1-to-clear. Reset: 0. This bit is set by a slave device whenever it terminates a cycle with target abort.
26:25	**DEVSELTiming: DEVSEL Timing**. Value: 1. These bits encode the timing of DEVSEL#. This module always responds in medium timing.
24	**DataParityErrorDetected: Data Parity Error Detected**. Read; Write-1-to-clear. Reset: 0. 1=The Parity Error Response bit is set and the module has detected PERR# asserted while acting as a PCI master regardless of whether PERR# was driven by this module.
23	**FastBack2BackCapable: Fast Back to Back Capable**. Value: 0. 0=Fast back to back cycles are not supported.
22	**UDFSupported: UDF Supported**. Value: 0. 0=User definable feature is not supported.
21	**Capable66MHz: 66 MHz Capable**. Value: 1. 1=This device is 66 MHz capable.
20	**MsiMappingCapability: MSI Mapping Capability**. Value: 0. 1=This device supports MSI mapping.
19:10	Reserved. Reset: 1.
9	**FastBack2BackEnable: Fast Back-to-Back Enable**. Value: 0. ACPI/SMBus does not support fast back-to-back.
8	**SERREnable: System Error Enable**. Read-write. Reset: 0. 1=Enable system error reporting.
7	**WaitCycleControl: Wait Cycle Control**. Value: 0. 0=This module does not use address stepping.

6	**ParityErrorResponse: Parity Error Response**. Read-write. Reset: 0. This bit controls the device's response to parity errors. 1=The device must take its normal action when a parity error is detected. 0=The device must ignore any parity errors that it detects and continue normal operation.
5	**VGAPaletteSnoop: VGA Palette Snoop**. Value: 0. This bit does not apply to this module.
4	**MemoryWriteInvalidateEnable: Memory Write and Invalidate Enable**. Value: 0. This module does not generate Memory Write and Invalidate commands.
3	**SpecialCycle: Special Cycle**. Value: 0. This module does not respond to special cycles.
2	**BusMaster: Bus Master**. Value: 0. ACPI/SMBus does not have a PCI master.
1	**MemorySpace: Memory Space**. Read-write. Reset: 1. 1=Enable the device to respond to memory space accesses.
0	**IOSpace: IO Space**. Read-write. Reset: 1. 1=Enable the device to respond to IO space accesses.

D14F0x08 Revision ID/Class Code

Bits	Description
31:8	**ClassCode: Class Code**. Value: C0500h. Specifies a SMBus controller.
7:0	**RevisionID: Revision ID**. Value: 4Ah.

D14F0x0C Cache Line Size

Bits	Description
31:24	**Bist**. Value: 0. The module has no built-in-self-test and so this is always 0.
23:16	**HeaderType: Header Type**. Value: 80h. This device is a multifunction device.
15:8	**LatencyTimer: Latency Timer**. Value: 0. This register specifies the value of the Latency Timer. This is not used in this module and so it is always 0.
7:0	**CacheLineSize: Cache Line Size**. Value: 0. Specifies the system cache line size. 0=Device does not use Memory Write and Invalidate commands and CacheLineSize is not applicable.

D14F0x10 Base Address 0

Bits	Description
31:0	**BaseAddress0: Base Address 0**. Value: 0.

D14F0x14 Base Address 1

Bits	Description
31:0	**BaseAddress1: Base Address 1**. Value: 0.

D14F0x18 Base Address 2

Bits	Description
31:0	**BaseAddress2: Base Address 2**. Value: 0.

D14F0x1C Base Address 3

Bits	Description
31:0	**BaseAddress3: Base Address 3**. Value: 0.

D14F0x20 Base Address 4

Bits	Description
31:0	**BaseAddress4: Base Address 4**. Value: 0.

D14F0x24 Base Address 5

Bits	Description
31:0	**BaseAddress5: Base Address 5**. Value: 0.

D14F0x28 Cardbus CIS Pointer

Bits	Description
31:0	**CardbusCISPointer: Cardbus CIS Pointer**. Value: 0.

D14F0x2C Subsystem Vendor ID

Bits	Description
31:16	**SubsystemID: Subsystem ID**. Write-once. Reset: 790Bh.
15:0	**SubsystemVendorID: Subsystem Vendor ID**. Write-once. Reset: 1022h.

D14F0x30 Expansion ROM Base Address

Bits	Description
31:8	Reserved.
7:0	**ExpansionROMBaseAddress: Expansion ROM Base Address**. Value: 0.

D14F0x34 Capability Pointer

Bits	Description
31:8	Reserved.
7:0	**CapabilityPointer: Capability Pointer**. Value: 0.

D14F0x3C Interrupt Line

Bits	Description
31:24	**MaxLat**. Value: 0. Device has no requirements for the setting.
23:16	**MinGnt**. Value: 0. Device has no requirements for the setting.
15:8	**InterruptPin: Interrupt Pin**. Value: 0. 0=This module does not generate interrupts.
7:0	**InterruptLine: Interrupt Line**. Value: 0. 0=This module does not generate interrupts.

D14F0xFC ScratchCfgReg

Bits	Description
31:0	**ScratchCfgReg: Scratch Configiure Register**. Read-write. Reset: 0. The usage is to be determined by software.

3.26.6.2 ASF (Alert Standard Format) Registers

ASF register space is accessed through two methods:
- IO access through ASF IO base address:
 A. Program the base address of ASF IO space through PMx00[SmbusAsfIoBase].
 B. Enable ASF IO decoding through PMx00[SmbusAsfIoEn].
- Direct memory mapped access through the AcpiMmio region. The ASF registers range from FED8_0000h+900h to FED8_0000h+9FFh. See PMx04[MmioEn].

ASFx00 HostStatus

Bits	Description
7	**LastByte**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 0=Last byte has not received. 1=Last byte has received.
6:5	Reserved.
4	**PECError**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 0=No CRC error. 1=CRC error happened.
3	**BusCollision**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 0=No bus collision. 1=Bus collision.
2	**DevError**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 0=Slave device behaves correctly. 1=No ACK or slave device responses incorrectly.

1	**Intr**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Termination of a command. Whenever [HostBusy] is cleared, this bit is set.
0	**HostBusy**. Read-only; Updated-by-hardware. Reset: 0. This bit reports the status of the ASF host. 0=SM bus host is idle. 1=SM bus host is busy.

ASFx02 HostControl

Bits	Description
7	**PECEnable**. Read-write. Reset: 0. 0=PEC disable. 1=PEC enable. Enable CRC checking when ASF HC presents as SM master and SM slave.
6	**Start**. Read; Write-1-only. Reset: 0. 1=Writing 1 to initiate the command. Always read back as 0.
5	**PECAppend**. Read-write. Reset: 0. 0=No PEC append. 1=Automatic PEC append. ASF HC calculates CRC code and append to the tail of the data packets.
4:2	**Protocol**. Read-write. Reset: 0. Bits Definition 000b Quick 001b Byte 010b Byte data 011b Word data 100b Process call 101b Block 110b Block write-block read-process call 111b Reserved
1	**KillHost**. Read-write. Reset: 0. 0=Enable SM master. 1=Reset SM master.
0	Reserved.

ASFx03 HostCommand

Bits	Description
7:0	**HostCommand**. Read-write. Reset: 0. Command to be transmitted by master.

ASFx04 SlaveAddress

Bits	Description
7:1	**Address**. Read-write. Reset: 0. Provide the SM address of slave.
0	**RW**. Read-write. Reset: 0. 0=Write. 1=Read.

ASFx05 Data0

Bits	Description
7:0	**Data0**. Read-write. Reset: 0. Contains count or DATA0 field of transaction.

ASFx06 Data1

Bits	Description
7:0	**Data1**. Read-write. Reset: 0. Contains DATA1 field of transaction.

ASFx07 DataIndex

Bits	Description
7:0	**DataIndex**. Read-write. Reset: 0. It is mapped to 72 data registers in data buffer for block write/read and block write-block read-process call.

ASFx08 PEC

Bits	Description
7:0	**Pec**. Read-write. Reset: 0. PEC byte to be sent to slave.

ASFx09 ListenAdr

Bits	Description
7:1	**ListenAdr**. Read-write. Reset: 0. The slave address which ASF slave responds in listen mode.
0	**ListenAdrEn**. Read-write. Reset: 010_0000b. 1=Enable Listen Mode when the slave address equals to ListenAdr[7:1]. 0=Disable Listen Mode when the slave address equals to ListenAdr[7:1].

ASFx0A ASFStatus

Bits	Description
7:6	**SlaveBusy**. Read-Only. Reset: 0. Indicates if ASF slave is receiving data.
5	**SlaveIntr**. Read; Write-1-to-clear. Reset: 0. ASF Slave interrupt Status.
4	Reserved.
3	**RemotePowerCycle**. Read; Write-1-to-clear. Reset: 0. 1=Power cycle has been triggered by ASF. 0=No power cycle ASF event.
2	**RemotePowerUp**. Read; Write-1-to-clear. Reset: 0. 1=Power up has been triggered by ASF. 0=No Power up ASF event.
1	**RemotePowerDown**. Read; Write-1-to-clear. Reset: 0. 1=Power down has been triggered by ASF. 0=No Power down ASF event.
0	**RemoteReset**. Read; Write-1-to-clear. Reset: 0. 1=Reset has been triggered by ASF. 0=No Reset cycle ASF event.

ASFx0B StatusMask0

Bits	Description
7:6	Reserved. Reset: 11b.
5	**FanSpeed0StatusEnable**. Read-write. Reset: 1. 1=Report Fan0 Speed Status to ASF. 0=No report.
4:1	Reserved. Reset: 1111b.
0	**Temp0StatusEnable**. Read-write. Reset: 1. 1=Report TempTsi status to ASF. 0=No report.

ASFx0C StatusMask1

Bits	Description
7:0	Reserved. Reset: FFh.

ASFx0D SlaveStatus

Bits	Description
7:4	Reserved.
3	**WrongSP**. Read; Write-1-to-clear. Reset: 0. 1=No SP symbol is detected when bus turns to read.
2	**SlaveDevError**. Read; Write-1-to-clear. Reset: 0. 1=Unexpected response.
1	**SlaveBusCollision**. Read; Write-1-to-clear. Reset: 0. 1=BusCollision happens.
0	**SlavePECError**. Read; Write-1-to-clear. Reset: 0. 1=PEC error.

ASFx0E RemoteCtrlAdr

Bits	Description
7:1	**RemoteCtrlAdr**. Read-write. Reset: 54h. SM address of Remote Control device.
0	Reserved.

ASFx0F SensorAdr

Bits	Description
7:1	**SensorAdr**. Read-write. Reset: 55h. SM address of Sensor.
0	Reserved.

ASFx10 DataReadPointer

Bits	Description
7:0	**DataReadPointer**. Read-only. Reset: 0. Current read pointer to the value specified in this register.

ASFx11 DataWritePointer

Bits	Description
7:0	**DataWritePointer**. Read-only. Reset: 0. Show current write pointer to the value specified in this register.

ASFx12 SetDataReadPointer

Bits	Description
7:0	**SetDataReadPointer**. Read-write. Reset: 0. Force the current write pointer to the value specified in this register.

ASFx13 DataBankSel

Bits	Description
7	**SetReadHostDataBank**. Read-write. Reset: 1. 1=Select to read data from Host Data Bank. 0=Select to read data from Data Bank 0 or Data Bank 1 decided by SetReadRevDataBank.
6	Reserved.
5:4	**SetReadRevDataBank**. Read-write. Reset: 0. Bits — Definition 00b — Select to read data from Data Bank 0. 01b — Select to read data from Data Bank 1. 10b — Select to read data from Data Bank 1. 11b — Select to read data from Data Bank 1.
3	**Databank1Full**. Read; Write-1-to-clear. Reset: 0. 0=Data Bank 1 is free. 1=Data Bank 1 is full.
2	**Databank0Full**. Read; Write-1-to-clear. Reset: 0. 0=Data Bank 0 is free. 1=Data Bank 0 is full.
1	**DataBank[1]**. Read; Write-1-to-clear. Reset: 0. 0=Data Bank still has space. 1=Data Bank is now full.
0	**DataBank[0]**. Read; Write-1-to-clear. Reset: 1. 0=Data Bank 0 is the latest touched data bank. 1=Data Bank 1 is the latest touched data bank.

ASFx14 Semaphore

Bits	Description
7:4	Reserved.
3	**ClrEcSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Write 1 to clear [EcSemaphore] bit.
2	**EcSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. This bit can only be set when [HostSemaphore] is clear.

1	**ClrHostSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Write 1 to clear [HostSemaphore] bit.
0	**HostSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Bits[0] and [2] are meant to be used as software semaphore between the host and Integrated Micro-Controller (IMC). When both host and IMC want to use the same resource, they can write to these semaphore bits first, followed by a read. If the read returns a 1 in the semaphore bit, it means it has established the semaphore first. This bit can only be set when EcSemaphore is clear.

ASFx15 SlaveEn

Bits	Description
7	**FairArbEn**. Read-write. Reset: 0. 0=Disable Fair Arbiter logic. 1=Enable Fair Arbiter logic, which forces ASF master to give up SMBus for a certain time, specified in register at offset 16h.
6	**TmrOutEn**. Read-write. Reset: 1. 1=Enable timer out function.
5	**LegacySensorEn**. Read-write. Reset: 0. 1=Enable Legacy Sensor.
4	**KillSlave**. Read-write. Reset: 0. Write 1 to reset Slave ASF Slave state machine.
3	**SuspendSlave**. Read-write. Reset: 0. Write 1 to Suspend (stop) ASF Slave state machine.
2	Reserved.
1	**SlaveIntrListenEn**. Read-write. Reset: 0. 1=Allow ASF slave to generate slave interrupt when the address of received packet is the same as the one specified in ListenAdr register ASFx09 [ListenAdr].
0	Reserved.

ASFx16 DelayMasterTimer

Bits	Description
7:0	**FairArbTimer**. Read-write. Reset: 04h. Specify how long ASF master has to wait before submitting next packet. Wait time = FairArbTimer * 2 us.

3.26.6.3 SMBus Registers

SMBus register space is accessed through two methods:

- IO access through SMBus IO base address:
 A. Program the base address of SMBus IO space through PMx00[SmbusAsfIoBase].
 B. Enable SMBus IO decoding through PMx00[SmbusAsfIoEn].
- Direct memory mapped access through the AcpiMmio region. The SMBus registers range from FED8_0000h+A00h to FED8_0000h+AFFh. See PMx04[MmioEn].

SMBUSx00 SMBusStatus

Bits	Description
7:5	Reserved.
4	**Failed**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. 1=A Failed bus transaction. IF (SMBUSx02[Kill] == 1) THEN SMBUSx00[Failed] = 1. ENDIF.
3	**BusCollision**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. 1=An SMBus transaction collision.

2	**DeviceErr**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. 1=An error of one of the following occurred: • Illegal command field. • Unclaimed cycle • Host device time-out.
1	**SMBusInterrupt**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. 1=Completion of the last host command.
0	**HostBusy**. Read-only. Reset: 0. 1=The SMBus controller is in the process of completing a command. When this bit is set, software should not access any other SMBus registers.

SMBUSx01 SMBusSlaveStatus

Bits	Description
7:6	Reserved.
5	**AlertStatus**. Read-only. Reset: 0.
4	**Shadow2Status**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. This bit is set by hardware to indicate a slave cycle address match of the SMB_Shadow2 port.
3	**Shadow1Status**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. This bit is set by hardware to indicate a slave cycle address match of the SMB_Shadow1 port.
2	**SlaveStatus**. Read; Write-1-to-clear; Set-by-hardware. Reset: 0. This bit is set by hardware to indicate a slave cycle event match of the SMBus slave command and SMBus Slave Event match.
1	**SlaveInit**. RAZ; Write-1-only. Reset: 0. Writing a 1 to this bit initializes the slave.
0	**SlaveBusy**. Read-only. Reset: 0. This bit indicates the SMBus controller slave interface is in the process of receiving data. Software should not try to access any other SMBus register when this bit is set.

SMBUSx02 SMBusControl

Bits	Description
7	**Reset**. Write-only. Reset: 0. Set the bit to 1 to stop SMBus transaction and reset SMBus controller state machine.
6	**Start**. Read-write. Reset: 0. Writing a 1 in this field initiates SMBus controller host interface to execute the command programmed in the [SMBusProtocol] field.
5	Reserved.
4:2	**SMBusProtocol**. Read-write. Reset: 0. Bits — Definition 000b — Quick Read or Write 001b — Byte Read or Write 010b — Byte Data Read or Write 011b — Word Data Read or Write 100b — Reserved 101b — Block Read or Write 111b-110b — Reserved

1	**Kill**. Read-write. Reset: 0. Stop the current host transaction in process.
0	**InterruptEnable**. Read-write. Reset: 0. 1=Enable the generation of interrupts on the completion of current host transaction.

SMBUSx03 SMBusHostCmd

Bits	Description
7:0	**SMBusHostCmd**. Read-write. Reset: 0. This field contains the data transmitted in the command field of SMBus host transaction.

SMBUSx04 SMBusAddress

Bits	Description
7:1	**SMBusAddr**. Read-write. Reset: 0. This field contains the 7-bit address of the target slave device.
0	**SMBusRdWr**. Read-write. Reset: 0. 0=Execute a Write command. 1=Execute a Read command.

SMBUSx05 SMBusData0

Bits	Description
7:0	**SMBusData0**. Read-write. Reset: 0. This register should be programmed with a value to be transmitted in the data 0 field of an SMBus host interface transaction. For block write commands, the count of the memory should be stored in this field. The value of this register is loaded into the block transfer count field. A valid value for block command count is between 1 and 32. For block reads, count received from SMBus device is stored here.

SMBUSx06 SMBusData1

Bits	Description
7:0	**SMBusData1**. Read-write. Reset: 0. This register should be programmed with a value to be transmitted in the data 1 field of an SMBus host interface transaction. When (SMBUSx14[SMBusPoll2Byte] == 1) & ((SMBUSx02[SMBusProtocol] == 001b) \| (SMBUSx02[SMBusProtocol] == 010b), this register is used as the data field in the second transaction.

SMBUSx07 SMBusBlockData

Bits	Description
7:0	**SMBusBlockData**. Read-write. Reset: 0. This register is used to transfer data into or out of the block data storage array.

SMBUSx08 SMBusSlaveControl

Bits	Description
7	**ClrEcSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Write 1 to clear EcSemaphore bit.
6	**EcSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. This bit can only be set when HostSemaphore is clear. Reading returns the value of this bit.
5	**ClrHostSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Write 1 to clear HostSemaphore bit.
4	**HostSemaphore**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. HostSemaphore and EcSemaphore are meant to be used as software semaphore between the host and embedded controller. When both host and Integrated Micro-Controller (IMC) want to use the same resource, they can write to these semaphore bits first, followed by a read. If the read returns a 1 in the semaphore bit, it means it has established the semaphore first. Write 1 to set this bit. This bit can only be set when EcSemaphore is clear. Reading returns the value of this bit.
3	**SMBusAlertEnable**. Read-only. Reset: 0.
2	**SMBusShadow2En**. Read-write. Reset: 0. Enable the generation of an interrupt or resume event upon an external SMBus master generating a transaction with an address that matches the SMBus Shadow 2 register.
1	**SMBusShadow1En**. Read-write. Reset: 0. Enable the generation of an interrupt or resume event upon an external SMBus master generating a transaction with an address that matches the SMBus Shadow 1 register.
0	**SlaveEnable**. Read-write. Reset: 0. Enable the generation of an interrupt or resume event upon an external SMBus master generating a transaction with an address that matches the host controller slave port of 10h, a command field that matches the SMBus slave control register, and a match of corresponding enabled events.

SMBUSx09 SMBusShadowCmd

Bits	Description
7:0	**SMBusShadowCmd**. Read-write. Reset: 0. This field contains the command value that was received during an external SMBus master access whose address field matched the host slave address (10h) or one of the slave shadow ports.

SMBUSx0A SMBusSlaveEvent

Bits	Description
15:0	**SMBusSlaveEvent**. Read-write. Reset: 0. This field contains data bits used to compare against incoming data to the SMBus Slave Data register (SMBUSx0C). When a bit in this register is 1 and a corresponding bit in SMBus slave's register is set, then an interrupt or resume event is generated if the command value matches the value in the SMBus slave's control register and the access was to SMBus host address 10h.

SMBUSx0C SlaveData

Bits	Description
15:0	**SlaveData**. Read-write. Reset: 0. This field contains the data value which was transmitted during an external SMBus master access whose address field matched one of the slave shadow port addresses or the SMBus host controller slave port address of 10h.

SMBUSx0E SMBusTiming

Bits	Description
7:0	**SMBusTiming**. Read-write. Reset: B0h. This register controls the frequency on the SMBus. The formula to calculate the frequency is: Frequency = 66 MHz/(SMBusTiming * 4).

SMBUSx10 I2CbusConfig

Bits	Description
7:4	**I2CRevision**. Read-write. Reset: 0. SMBus controller revision.
3:1	Reserved.
0	**I2CbusInterrupt**. Read-write. Reset: 0. 0=SMI#. 1=IRQ.

SMBUSx11 I2CCommand

Bits	Description
7:0	**I2Ccommand**. Read-write. Reset: 0. I^2C Host Slave Command. This value specifies the command value to be matched for I^2C master accesses to the I^2C controller host slave interface.

SMBUSx12 I2CShadow1

Bits	Description
7:1	**I2CslaveAddr1**. Read-write. Reset: 0. SMBus Slave Address for shadow port 1. This value specifies the address used to match against incoming I^2C addresses for Shadow port 1.
0	**ReadWriteShadowPort1**. Read-write. Reset: 0. This bit must be programmed to 0 because the I^2C slave controller only responds to Word Write Transactions. 0=Write command. 1=Read command.

SMBUSx13 I2Cshadow2

Bits	Description
7:1	**I2CslaveAddr2**. Read-write. Reset: 0. SMBus slave address for shadow port 2. This value specifies the address used to match against incoming I^2C addresses for Shadow port 2.
0	**ReadWriteShadowPort2**. Read-write. Reset: 0. This bit must be programmed to 0 because I^2C slave controller only responds to Word Write Transaction. 0=Write command. 1=Read command.

SMBUSx14 SMBusAutoPoll

Bits	Description
7	**SMBusPoll2Byte**. Read-write. Reset: 1. 1=IF ((SMBUSx02[SMBusProtocol] == 001b) \|\| (SMBUSx02[SMBusProtocol] == 010b)) THEN the last command will be performed twice at each polling interval. SMBUSx06 contains data transmitted and retrieved from the second transaction. SMBUSx17 contains data transmitted in the command field in the second transaction.
6:4	Reserved.
3:1	**SMBusPollPeriod**. Read-write. Reset: 2h. Bits — Definition 000b — poll every 1/16 second 001b — poll every 1/8 second 010b — poll every 1/ 4 second 011b — poll every 1/ 2 second 111b-100b — poll every 1 second
0	**SMBusAutoPollEn**. Read-write. Reset: 0. 1=SMBus will periodically execute the last command whenever SMBUSx15 expires. The polling interval is defined by [SMBusPollPeriod]. The purpose of this function is to use the SMBus to read CPU temperature. Values in SMBUSx05 and SMBUSx06 are to be used for controlling the fan speed.

SMBUSx15 SMBusCounter

Bits	Description
7:0	**SMBusCounter**. Read-only; Updated-by-hardware. This counter specifies the remaining time until the next read specified as (SMBusCounter*SMBUSx14[SMBusPollPeriod]/256) us.

SMBUSx16 SMBusStop

Bits	Description
15:1	Reserved.
0	**SMBusPausePoll**. Read-write. Reset: 0. This is to be used as a semaphore mechanism by software to stop the next polling. If software wants to use the SMBus for other purposes but the hardware has already been enabled (SMBUSx14[SMBusAutoPollEn] == 1) to perform auto polling, software can halt the next polling by setting this bit to 1. If the halt is successful, this bit will return 1. If the hardware has already started the polling operation at the same time software tries to set this bit, this bit returns 0. When the pending polling operation is finally complete, this bit then get sets. When this bit is set, polling counter temporarily stops (not resets). It is now safe for software to access the SMBus. Upon completion, software must restore the previous command in the appropriate register prior to setting this bit. software should also clear this bit so the counter can resume counting.

SMBUSx17 SMBusHostCmd2

Bits	Description
7:0	**SMBusHostCmd2**. Read-write. Reset: 0. This field contains the data transmitted in the command field in the second command when (SMBUSx14[SMBusPoll2Byte] == 1) and (SMBUSx02[SMBus-Protocol] == 010b).

3.26.7 IOAPIC Registers

IOAPIC configuration registers can be accessed at memory region from FEC0_0000h to FEC0_007Fh. Program PMx00[IoApicEn] = 1 to enable IOAPIC decoding.

IOAPICx00 IO Register Select Register

Bits	Description
31:8	Reserved.
7:0	**IndirectAddressOffset**. Read-write. Reset: 0. Indirect Address Offset to IO Window Register. It is used to determine which register is manipulated during an IO Window Register read/write operation.

IOAPICx10 IO Window Register

Bits	Description
31:0	**IOWindow**. Read-write. Reset: 0. Mapped by IOAPICx00 to the designated indirect access register.

IOAPICx10_x00 IOAPIC ID Register

This register is not used in IOxAPIC PCI bus delivery mode.

Bits	Description
31:28	Reserved.
27:24	**ID**. Read-write. Reset: 0. IOAPIC device ID.
23:0	Reserved.

IOAPICx10_x01 IOAPIC Version Register

Bits	Description
31:24	Reserved.
23:16	**MaxRedirectionEntries**. Value: 17h. Indicates 24 entries [23:0].
15	**PRQ**. Value: 1. IRQ pin assertion supported.
14:8	Reserved.
7:0	**Version**. Value: 21h. PCI 2.2 compliant.

IOAPICx10_x02 IOAPIC Arbitration Register

This register is not used in IOxAPIC PCI bus delivery mode.

Bits	Description
31:28	Reserved.
27:24	**ArbitrationID: Arbitration ID**. Read-only. Reset: 0.
23:0	Reserved.

IOAPICx10_x[3E:10:step2] Redirection Table Entry [23:0]

Bits	Description
63:56	**DestinationID**. Read-write. Reset: 0. Bits [19:12] of the address field of the interrupt message.
55:32	Reserved.
31:17	Reserved.
16	**Mask**. Read-write. Reset: 1. 1=Mask the interrupt injection at the input of this device. 0=Unmask.
15	**TriggerMode**. Read-write. Reset: 0. 0=Edge. 1=Level.
14	**RemoteIRR**. Read-only. Reset: 0. Used for level triggered interrupts only. It is cleared by EOI special cycle transaction or write to EOI register. 1=Interrupt message is delivered.
13	**InterruptPinPolarity**. Read-write. Reset: 0. 0=High. 1=Low.
12	**DeliveryStatus**. Read-only. Reset: 0. 0=Idle. 1=Send Pending.
11	**DestinationMode**. Read-write. Reset: 0. 0=Physical. 1=Logical.
10:8	**DeliveryMode**. Read-write. Reset: 0. Bits Definition Bits Definition 000b Fixed 100b NMI 001b Lowest Priority 101b INIT 010b SMI/PMI 110b Reserved 011b Reserved 111b ExtINT
7:0	**Vector**. Read-write. Reset: 0. Interrupt vector associated with this interrupt input.

IOAPICx20 IRQ Pin Assertion Register

Bits	Description
31:8	Reserved.
7:0	**InputIrq**. Read-write. Reset: 0. IRQ number for the requested interrupt. A write to this register triggers an interrupt associated with the redirection table entry referenced by the IRQ number. Currently the redirection table has 24 entries. Writes with IRQ number greater than 17h have no effect.

IOAPICx40 EOI Register

Bits	Description
31:8	Reserved.
7:0	**Vector**. Write-only. Reset: 0. Interrupt vector. A write to this register clears the remote IRR bit in the redirection table entry found matching the interrupt vector. This provides an alternate mechanism other than PCI special cycle for EOI to reach IOxAPIC.

3.26.8 LPC-ISA Bridge

3.26.8.1 Device 14h Function 3 (LPC Bridge) Configuration Registers

See 3.1 [Register Descriptions and Mnemonics] for a description of the register naming convention. See 2.7 [Configuration Space] for details about how to access this space.

D14F3x00 Device/Vendor ID

Bits	Description
31:16	**DeviceID: Device Identifier**. Value: 790Eh. This 16-bit field is assigned by the device manufacturer and identifies the type of device.
15:0	**VendorID: Vendor Identifier**. Read-only. Reset: 1022h.

D14F3x04 Status/Command

Bits	Description
31	**DetectedParityError: Detected Parity Error**. Read; Write-1-to-clear. Reset: 0. 1=The FCH detects a parity error.
30	**SignaledSystemError: Signaled System Error**. Read; Write-1-to-clear. Reset: 0. 1=The FCH detects a PCI address parity error.
29	**ReceivedMasterAbort: Received Master Abort**. Read; Write-1-to-clear. Reset: 0. 1=The FCH acts as a PCI master and aborts a PCI bus memory cycle.
28	**ReceivedTargetAbort: Received Target Abort**. Read; Write-1-to-clear. Reset: 0. 1=An FCH generated PCI cycle (the FCH is the PCI master) is aborted by a PCI target.
27	**SignaledTargetAbort: Signaled Target Abort**. Read; Write-1-to-clear. Reset: 0. 1=The FCH signals target abort.
26:25	**DeviceSelectTiming: Device Select Timing**. Value: 1. Indicates DEVSEL# timing when performing a positive decode. DEVSEL# is asserted to meet the medium timing.
24	**MasterDataParityError: Master Data Parity Error**. Read; Write-1-to-clear. Reset: 0. 1=The FCH detects PERR# asserted while acting as PCI master regardless whether PERR# was driven by the FCH or not.
23:21	Reserved.
20	**CapabilitiesList: Capabilities List**. Read-only. Reset: 0.
19:10	Reserved.
9	**FastBack2BackEnable: Fast Back-to-Back Enable**. Value: 0.
8	**SERREnable: SERR# Enable**. Read-only. Reset: 0. 1=The FCH asserts SERR# when it detects an address parity error. 0=SERR# is not asserted.
7	**SteppingControl: Stepping Control**. Value: 0.
6	**ParityErrorResponse: Parity Error Response**. Read-write. Reset: 0. PERR# (Response) Detection Enable bit. 1=The FCH asserts PERR# when it is the agent receiving data and it detects a parity error. 0=PERR# is not asserted.
5	**VGAPaletteSnoop: VGA Palette Snoop**. Value: 0.
4	**MemoryWriteAndInvalidateEnable: Memory Write and Invalidate Enable**. Value: 0.

3	**SpecialCycles: Special Cycles**. Value: 1.
2	**BusMaster: Bus Master**. Read-only. Reset: 1. 1=Bus master enabled.
1	**MemorySpace: Memory Space**. Value: 1.
0	**IOSpace: IO Space**. Read-only. Reset: 1. This bit controls access to the IO space registers. 1=Enable access to the legacy IDE ports, and PCI bus master IDE IO registers are enabled.

D14F3x08 Revision ID/Class Code

Bits	Description
31:8	**ClassCode: Class Code**. Read-only. Reset: 0601_00h. Indicates an ISA bridge.
7:0	**RevisionID: Revision ID**. Value: 1. Indicates the revision level of the chip design.

D14F3x0C Cache Line Size

Bits	Description
31:24	**Bist**. Read-only. Reset: 0. No BIST modes.
23:16	**HeaderType: Header Type**. Read-only. Reset: 80h. Identifies the type of the predefined header in the configuration space. The most significant bit is set to 1 to indicate a multifunction device.
15:8	**LatencyTimer: Latency Timer**. Read-only. Reset: 0. Specifies the value of the latency timer in units of PCICLKs.
7:0	**CacheLineSize: Cache Line Size**. Value: 0.

D14F3x10 Base Address Reg 0

Bits	Description
31:5	**BaseAddress0[31:5]: Base Address 0**. RAZ; Write-only. Reset: 7F60_000h. This register has an internal value used as base address for APIC memory space. Writing to the register changes its internal value. The default internal base address is FEC0_0000h.
4:0	**BaseAddress0[4:0]**. Value: 0.

D14F3x2C Subsystem ID and Subsystem Vendor ID

Bits	Description
31:16	**SubsystemID: Subsystem ID**. Read; Write-once. Reset: 780Eh.
15:0	**SubsystemVendorID: Subsystem Vendor ID**. Read; Write-once. Reset: 1022h.

D14F3x34 Capabilities Pointer

Bits	Description
31:8	Reserved.
7:0	**CapabilitiesPointer: Capabilities Pointer**. Read-only. Reset: 0.

D14F3x40 PCI Control

Bits	Description
31:8	Reserved.
7	**IntegratedEcPresent**. Read-only; Updated-by-hardware. Reset: 0. 1=Integrated Micro-Controller (IMC) is present.
6	**EcSemaphore**. Read-write. Reset: 0. This bit is writable by IMC and read by BIOS. This is used as the software semaphore mechanism between BIOS and IMC to see who can access the common resource. IMC should read bit [BiosSemaphore] first to see if BIOS has taken ownership of the resource. If [BiosSemaphore] == 0, then IMC should write a 1 to this bit and then follow by a read to see if this bit is set. If this bit is set, it means IMC has successfully taken ownership of the resource. If this bit returns a 0 and [BiosSemaphore] returns a 1, then BIOS has taken ownership first. IMC should always clear this bit after it has completed its access to the resource.
5	**BiosSemaphore**. Read-write. Reset: 0. This bit is writable by BIOS and read by the IMC. This is used as the software semaphore mechanism between BIOS and IMC to see who can access the common resource. BIOS should read bit [EcSemaphore] first to see if IMC has taken ownership of the resource. If [EcSemaphore] == 0, then BIOS should write a 1 to this bit and then follow by a read to see if this bit is set. If this bit is set, it means BIOS has successfully taken ownership of the resource. If this bit returns 0 and [EcSemaphore] returns a 1, then IMC has taken ownership first. Software should always clear this bit after it has completed its access to the resource.
4:3	Reserved.
2	**LegacyDmaEnable: Legacy DMA Enable**. Read-write. Reset: 1. BIOS: See 2.17.4.1. 1=Enable LPC DMA cycle. Transfer size for channel 3-0 is 8 bits; Transfer size for channel 7-5 is 16 bits. 32-bit DMA is not supported.
1:0	Reserved.

D14F3x44 IO Port Decode Enable

Bits	Description
31	**AdLibPortEnable: Ad-Lib Port Enable**. Read-write. Reset: 0. Port enable for Ad-Lib port, 388h-389h. 1=Enable the IO range.
30	**AcpiMicroControllerPortEnable: ACPI Micro-Controller Port Enable**. Read-write. Reset: 0. Port enable for ACPI micro-controller port, 62h and 66h. 1=Enable the IO range.
29	**KBCPortEnable: KBC Port Enable**. Read-write. Reset: 0. Port enable for KBC port, 60h and 64h. 1=Enable the IO range.
28	**GamePortEnable: Game Port Enable**. Read-write. Reset: 0. Port enable for game port, 200h-20Fh. 1=Enable the IO range.
27	**FDCPortEnable1: FDC Port Enable 1**. Read-write. Reset: 0. Port enable for FDC port, 370h-377h. 1=Enable the IO range.
26	**FDCPortEnable0: FDC Port Enable 0**. Read-write. Reset: 0. Port enable for FDC port, 3F0h-3F7h. 1=Enable the IO range.
25	**MSSPortEnable3: MSS Port Enable 3**. Read-write. Reset: 0. Port enable for MSS port, F40h-F47h. 1=Enable the IO range.

24	**MSSPortEnable2: MSS Port Enable 2**. Read-write. Reset: 0. Port enable for MSS port, E80h-E87h. 1=Enable the IO range.
23	**MSSPortEnable1: MSS Port Enable 1**. Read-write. Reset: 0. Port enable for MSS port, 604h-60bh. 1=Enable the IO range.
22	**MSSPortEnable0: MSS Port Enable 0**. Read-write. Reset: 0. Port enable for MSS port, 530h-537h. 1=Enable the IO range.
21	**MIDIPortEnable3: MIDI Port Enable 3**. Read-write. Reset: 0. Port enable for MIDI port, 330h-331h. 1=Enable the IO range.
20	**MIDIPortEnable2: MIDI Port Enable 2**. Read-write. Reset: 0. Port enable for MIDI port, 320h-321h. 1=Enable the IO range.
19	**MIDIPortEnable1: MIDI Port Enable 1**. Read-write. Reset: 0. Port enable for MIDI port, 310h-311h. 1=Enable the IO range.
18	**MIDIPortEnable0: MIDI Port Enable 0**. Read-write. Reset: 0. Port enable for MIDI port, 300h-301h. 1=Enable the IO range.
17	**AudioPortEnable3: Audio Port Enable 3**. Read-write. Reset: 0. Port enable for audio port, 280h-293h. 1=Enable the IO range.
16	**AudioPortEnable2: Audio Port Enable 2**. Read-write. Reset: 0. Port enable for audio port, 260h-273h. 1=Enable the IO range.
15	**AudioPortEnable1: Audio Port Enable 1**. Read-write. Reset: 0. Port enable for audio port, 240h-253h. 1=Enable the IO range.
14	**AudioPortEnable0: Audio Port Enable 0**. Read-write. Reset: 0. Port enable for audio port, 230h-233h. 1=Enable the IO range.
13	**SerialPortEnable7: Serial Port Enable 7**. Read-write. Reset: 0. Port enable for serial port, 3E8h-3EFh. 1=Enable the IO range.
12	**SerialPortEnable6: Serial Port Enable 6**. Read-write. Reset: 0. Port enable for serial port, 338h-33Fh. 1=Enable the IO range.
11	**SerialPortEnable5: Serial Port Enable 5**. Read-write. Reset: 0. Port enable for serial port, 2E8h-2EFh. 1=Enable the IO range.
10	**SerialPortEnable4: Serial Port Enable 4**. Read-write. Reset: 0. Port enable for serial port, 238h-23Fh. 1=Enable the IO range.
9	**SerialPortEnable3: Serial Port Enable 3**. Read-write. Reset: 0. Port enable for serial port, 228h-22Fh. 1=Enable the IO range.
8	**SerialPortEnable2: Serial Port Enable 2**. Read-write. Reset: 0. Port enable for serial port, 220h-227h. 1=Enable the IO range.
7	**SerialPortEnable1: Serial Port Enable 1**. Read-write. Reset: 0. Port enable for serial port, 2F8h-2FFh. 1=Enable the IO range.
6	**SerialPortEnable0: Serial Port Enable 0**. Read-write. Reset: 0. Port enable for serial port, 3F8h-3FFh. 1=Enable the IO range.
5	**ParallelPortEnable5: Parallel Port Enable 5**. Read-write. Reset: 0. Port enable for parallel port, 7BCh-7BFh. 1=Enable the IO range.
4	**ParallelPortEnable4: Parallel Port Enable 4**. Read-write. Reset: 0. Port enable for parallel port, 3BCh-3BFh. 1=Enable the IO range.
3	**ParallelPortEnable3: Parallel Port Enable 3**. Read-write. Reset: 0. Port enable for parallel port, 678h-67Fh. 1=Enable the IO range.

2	**ParallelPortEnable2: Parallel Port Enable 2**. Read-write. Reset: 0. Port enable for parallel port, 278h-27Fh. 1=Enable the IO range.
1	**ParallelPortEnable1: Parallel Port Enable 1**. Read-write. Reset: 0. Port enable for parallel port, 778h-77Fh. 1=Enable the IO range.
0	**ParallelPortEnable0: Parallel Port Enable 0**. Read-write. Reset: 0. Port enable for parallel port, 378h-37Fh. 1=Enable the IO range.

D14F3x48 IO or Memory Port Decode Enable

Bits	Description
31:26	Reserved.
25	**WideIO2Enable**. Read-write. Reset: 0. Port enable for wide generic IO port 2 defined by D14F3x90[IOBaseAddress2]. 1=Enable the IO range.
24	**WideIO1Enable**. Read-write. Reset: 0. Port enable for wide generic IO port 1 defined by D14F3x64[IOBaseAddress1]. 1=Enable the IO range.
23	**IOPortEnable6: IO port enable 6**. Read-write. Reset: 0. Port enable for IO port FD60h-FD6Fh. 1=Enable the IO range.
22	**IOPortEnable5: IO port enable 5**. Read-write. Reset: 0. Port enable for IO port 4700h-470Bh. 1=Enable the IO range.
21	**IOPortEnable4: IO port enable 4**. Read-write. Reset: 1. Port enable for IO port 80h. 1=Enable the IO range.
20	**MemPortEnable: Mem port enable**. Read-write. Reset: 0. Port enable for 4K byte memory range defined in D14F3x4C. 1=Enable the memory range.
19	**IOPortEnable3: IO port enable 3**. Read-write. Reset: 0. Port enable for IO port 580h-5BFh. 1=Enable the IO range.
18	**IOPortEnable2: IO port enable 2**. Read-write. Reset: 0. Port enable for IO port 500h-53Fh. 1=Enable the IO range.
17	**IOPortEnable1: IO port enable 1**. Read-write. Reset: 0. Port enable for IO port 480h-4BFh. 1=Enable the IO range.
16	**IOPortEnable0: IO port enable 0**. Read-write. Reset: 0. Port enable for IO port 400h-43Fh. 1=Enable the IO range.
15:8	**SyncTimeoutCount: Sync Timeout Count**. Read-write. Reset: FFh. When [SyncTimeoutCounterEnable] == 1, this is the number of LPC clocks that the state machine will wait during LPC data sync before aborting the cycle.
7	**SyncTimeoutCounterEnable: Sync Timeout Counter Enable**. Read-write. Reset: 0. 1=LPC sync timeout counter is enabled. 0=The counter is disabled. This counter is used to avoid a deadlock condition if an LPC device drives sync continuously. Timeout count is programmed in [SyncTimeoutCount]. Write 0 to this bit if an LPC device is extremely slow and takes more than 255 LPC clocks to complete a cycle.
6	**RtcIORangePortEnable: RTC IO Range Port Enable**. Read-write. Reset: 0. Port enable for RTC IO range 70h-73h. 1=Enable the IO range.
5	**MemoryRangePortEnable: Memory Range Port Enable**. Read-write. Reset: 0. Port enable for LPC memory target range defined by D14F3x60. 1=Enable the memory range.
4:3	Reserved.

2	**WideIO0Enable**. Read-write. Reset: 0. Port enable for wide generic IO port defined by D14F3x64[IOBaseAddress0]. 1=Enable the IO range.
1	**AlternateSuperIOConfigurationPortEnable: Alternate Super IO Configuration Port Enable**. Read-write. Reset: 0. Port enable for alternate Super IO configuration port, 4Eh-4Fh. 1=Enable the IO range.
0	**SuperIOConfigurationPortEnable: Super IO Configuration Port Enable**. Read-write. Reset: 0. Port enable for Super IO configuration port, 2Eh-2Fh. 1=Enable the IO range.

D14F3x4C Memory Range

Bits	Description
31:12	**BaseAddress: Base Address**. Read-write. Reset: 0. Specifies a 4K byte memory range from {Base Address, 000h} to {Base Address, FFFh}. The range is enabled by D14F3x48[MemPortEnable].
11:0	Reserved.

D14F3x[5C,58,54,50] ROM Protect 3, 2, 1, 0

These registers specify different ROM ranges to be protected. See 2.17.4.4 [Serial Peripheral Interface (SPI) ROM]. The addresses are within the defined ROM range if:
{RomBase, 000_0000_0000b} <= address[31:0] <= (({RomBase, 000_0000_0000b} + ((Range+1) << (RangeUnit ? 16 : 12))) - 1).
For host, these registers can only be written once after hardware reset; Subsequent writes to it have no effect.

Bits	Description
31:12	**RomBase: ROM Base**. Read; Write-once. Reset: 0.
11	Reserved.
10	**WriteProtect: Write Protect**. Read; Write-once. Reset: 0. 1=The memory range defined by this register is write-protected and writing to the range has no effect.
9	**ReadProtect: Read Protect**. Read; Write-once. Reset: 0. 1=The memory range defined by this register is read-protected and reading any location in the range returns FFFF_FFFFh.
8	**RangeUnit**. Read; Write-once. Reset: 0. 0=4-KB. 1=64-KB.
7:0	**Range**. Read; Write-once. Reset: 0. Specifies the protected range. The unit is defined by RangeUnit.

D14F3x60 PCI Memory Address for LPC Target Cycles

This register contains the upper 16 bits of the start and end address of the LPC memory target range. The lower 16 bits of MemoryStartAddress are 0000h. The lower 16 bits of MemoryEndAddress are FFFFh. This range can be enabled or disabled using D14F3x48[MemoryRangePortEnable].

Bits	Description
31:16	**MemoryEndAddress: Memory End Address**. Read-write. Reset: 0. Specifies the upper 16 bits of the end address of the LPC target memory range.
15:0	**MemoryStartAddress: Memory Start Address**. Read-write. Reset: 0. Specifies the upper 16 bits of the start address of the LPC target memory range.

D14F3x64 PCI IO base Address for Wide Generic Port

Bits	Description
31:16	**IOBaseAddress1: IO Base Address 1**. Read-write. Reset: 0. 16-bit PCI IO base address for wide generic IO port range. This function is enabled by D14F3x48[WideIO1Enable]. If D14F3x74[AlternativeWideIO1RangeEnable] == 1, the range is 16 bytes; else the range is 512 bytes.
15:0	**IOBaseAddress0: IO Base Address 0**. Read-write. Reset: 0. 16-bit PCI IO base address for wide generic IO port range. This function is enabled by D14F3x48[WideIO0Enable]. If D14F3x74[AlternativeWideIO0RangeEnable] == 1, the range is 16 bytes; else the range is 512 bytes.

D14F3x68 ROM Address Range 1

This register contains the upper 16 bits of the start and end address of the ROM address range 1. The lower 16 bits of RomStartAddress1 are 0000h. The lower 16 bits of the RomEndAddress1 are FFFFh. This range can be enabled or disabled using D14F3x48[RomRange1PortEnable]. This register is used for both LPC and SPI flash.

Bits	Description
31:16	**RomEndAddress1: ROM End Address 1**. Read-write. Reset: 0Fh. Specifies the upper 16 bits of the end address of the ROM memory address range 1.
15:0	**RomStartAddress1: ROM Start Address 1**. Read-write. Reset: 08h. Specifies the upper 16 bits of the start address of the ROM memory address range 1.

D14F3x6C ROM Address Range 2

This register contains the upper 16 bits of the start and end address of the ROM address range 2. The lower 16 bits of RomStartAddress2 are 0000h. The lower 16 bits of RomEndAddress2 are FFFFh. This range can be enabled or disabled using D14F3x48[RomRange2PortEnable]. This register is used for both LPC and SPI flash.

Bits	Description
31:16	**RomEndAddress2: ROM End Address 2**. Read-write. Reset: FFFFh. Specifies the upper 16 bits of the end address of the ROM memory address range 2.
15:0	**RomStartAddress2: ROM Start Address 2**. Read-write. Reset: FFF8h. Specifies the upper 16 bits of the start address of the ROM memory address range 2.

D14F3x74 Alternative Wide IO Range Enable

Bits	Description
31:4	Reserved.
3	**AlternativeWideIO2RangeEnable: Alternative Wide IO 2 Range Enable**. Read-write. Reset: 0. This bit is similar to bit[AlternativeWideIO0RangeEnable], but it applies to the IO range defined by D14F3x90[IOBaseAddress2]. See bit[AlternativeWideIO0RangeEnable] for detailed description.
2	**AlternativeWideIO1RangeEnable: Alternative Wide IO 1 Range Enable**. Read-write. Reset: 0. This bit is similar to bit[AlternativeWideIO0RangeEnable], but it applies to the IO range defined by D14F3x64[IOBaseAddress1]. See bit[AlternativeWideIO0RangeEnable] for detailed description.

1	Reserved.
0	**AlternativeWideIO0RangeEnable: Alternative Wide IO 0 Range Enable**. Read-write. Reset: 0. 0=Wide IO range defined by D14F3x64[IOBaseAddress0] is 512 bytes. 1=The range is 16 bytes. To use this feature, address in D14F3x64[IOBaseAddress0] must be aligned to 16 bytes (i.e., bits[3:0] must be 0). If the address is not aligned to 16 bytes, the IO range is from address[15:0] to {address[15:4], 0xF}.

D14F3x78 Miscellaneous Control Bits

Bits	Description
31:11	Reserved.
10	**LDRQ0_PD_EN**. Read-write. Reset: 1. 1=Enable the pull-down of LDRQ0 pad. 0=Disable the pull-down of LDRQ0 pad.
9	**LDRQ0_PU_EN**. Read-write. Reset: 0. 1=Enable the pull-up of LDRQ0 pad. 0=Disable the pull-up of LDRQ0 pad.
8	Reserved.
7	**AllowHostInDma**. Read-write. Reset: 1. 1=Allow Host to access LPC if ACPI has not given GNT to LPC during DMA transfer. 0=DMA hold LPC even ACPI has not given GNT to LPC during DMA transfer.
6	**GateWrongRx**. Read-write. Reset: 0. 1=Allow AltRxByteCount to be 0.
5	**GateSpiAccessDis**. Read-write. Reset: 0. 1=Pass ROM access to SPI even if it is strapped as LPC.
4	**SMMWriteRomEn**. Read-write. Reset: 1. 1=Enable ROM access in SMM mode.
3	**LDRQ1**. Read-write. Reset: 0. BIOS: See 2.17.4.1. 1=Enable LDRQ1# on LPC bus.
2	**LDRQ0**. Read-write. Reset: 0. BIOS: See 2.17.4.1. 1=Enable LDRQ0# on LPC bus.
1	Reserved.
0	**NoHog: No Hog**. Read-write. Reset: 1. BIOS: See 2.17.4.1. 1=The internal bus is not locked by LPC bridge during a slave access (e.g., LPC DMA fetch). 0=LPC may hold the internal bus during a DMA transfer.

D14F3x7C Trusted Platform Module (TPM)

Bits	Description
31:14	Reserved.
13	**TpmBufferEn**. Read-write. Reset: 0. 1=Enable TPM buffer. 0=Disable TPM buffer.
12	**TpmPfetchEn**. Read-write. Reset: 0. 1=Enable TPM burst read. 0=Disable TPM burst read.
11	**LpcClk1IsGpio**. Read-write. Reset: 1. 1=Treat LpcClk1 as GPIO. 0=Treat LpcClk1 as LpcClk1.
10	**GpioLpcClk1Out**. Read-write. Reset: 0. Control GpioLpcClk1 output value.
9	**GpioLpcClk1OeB**. Read-write. Reset: 1. 1=Disable GpioLpcClk1 output. 0=Enable GpioLpcClk1 output.
8	**GpioLpcClk1**. Read-only. Reset: X. Status of LpcClk1 port.
7	**WiderTpmEn**. Read-write. Reset: 0. 1=Force logic to decode FED4_XXXXh as TPM cycles instead of FED4_0XXXh, FED4_1XXXh, FED4_2XXXh, FED4_3XXXh, and FED4_4XXXh.

6:5	**TmkbcSel**. Read-write. Reset: 0. Select which one of the four sets of TMKBC registers specified in D14F3x84, D14F3x88, D14F3x8C are accessed. Bits Definition 00b set 0 01b set 1 10b set 2 11b set 3
4	**TmkbcSet**. Write-once. Reset: 0. 1=All Trusted Mobile Keyboard Controller (TMKBC) address/remap registers cannot be changed until the next reset.
3	**TmkbcEnable**. Read-write. Reset: 0. 1=Enable the TMKBC function.
2	**TpmLegacy**. Read-write. Reset: 0. 1=Enable decoding of legacy TPM addresses: IO addresses 7Fh-7Eh and EFh-EEh.
1	Reserved.
0	**Tpm12En**. Read-write. Reset: 0. 1=Enable decoding of TPM cycles defined in TPM 1.2 spec. Note that this bit and [TpmLegacy] are independent bits; they respectively turn on decoding of different TPM addresses.

D14F3x84 TMKBC_BaseAddrLow

Bits	Description
31:7	**TmkbcBaseAddrLow**. Read-write. Reset: 0. This register defines the lower 32-bit memory address used for the TMKBC function. There are actually four sets of such mapping. The selection is controlled by D14F3x7C[TmkbcSel].
6	**MaskBits13thru8**. Read-write. Reset: 0. Defines whether TMKBC address bits[13:8] are masked as do-not-care bits. 1=Masked. 0=No mask.
5	**MaskBits12thru8**. Read-write. Reset: 0. Defines whether TMKBC address bits[12:8] are masked as do-not-care bits. 1=Masked. 0=No mask.
4	**MaskBits11thru8**. Read-write. Reset: 0. Defines whether TMKBC address bits[11:8] are masked as do-not-care bits. 1=Masked. 0=No mask.
3	**MaskBits10thru8**. Read-write. Reset: 0. Defines whether TMKBC address bits[10:8] are masked as do-not-care bits. 1=Masked. 0=No mask.
2	**Addr64**. Read-write. Reset: 0. Defines whether the TMKBC address is 32 or 64 bits. 1=The address is 64 bits. 0=The address is 32 bits.
1:0	Reserved.

D14F3x88 TMKBC_BaseAddrHigh

Bits	Description
31:0	**TmkbcBaseAddrHigh**. Read-write. Reset: 0. If D14F3x84[Addr64] == 1, this register defines the upper 32-bit memory address used for the TMKBC function. If D14F3x84[Addr64] == 0, this register has no meaning. There are four sets of such mapping. The selection is controlled by D14F3x7C[TmkbcSel].

D14F3x8C TMKBC_Remap

Bits	Description
31:16	Reserved.
15:8	**TmkbcRemap**. Read-write. Reset: 0. This register defines the remap address[15:8] on the LPC bus. There are four sets of such mapping. The selection is controlled by D14F3x7C[TmkbcSel].
7:0	Reserved.

D14F3x90 Wide IO 2

Bits	Description
31:16	Reserved.
15:0	**IOBaseAddress2: IO Base Address 2**. Read-write. Reset: 0. 16-bit PCI IO base address for wide generic IO port range. This function is enabled by D14F3x48[WideIO2Enable]. If D14F3x74[AlternativeWideIO2RangeEnable] == 1, the range is 16 bytes; else, the ranges is 512 bytes.

D14F3x98 EC_LPC_Cntrl

Bits	Description
31:9	Reserved.
8	**EcHoldLpc**. Read-write. Reset: 0. 1=IMC holds the LPC bridge and host cannot access LPC.
7	**ArbiterParkEn**. Read-write. Reset: 0. 1=Enable UsbReq park at grant state.
6:1	Reserved.
0	**HostHoldLpc**. Read-write. Reset: 0. 1=Host holds the LPC bridge and prevents IMC from accessing LPC.

D14F3xA0 SPI Base_Addr

Bits	Description
31:6	**Spi_eSpi_BaseAddr[31:6]**. Read-write. Reset: 0. BIOS: See 2.17.4 [LPC Bus Interface]. This register specifies the MMIO base address for the SPI ROM controller and eSPI host controller registers. SPI BAR={Spi_eSpi_BaseAddr[31:6], 000000b}. See 3.26.8.2 [SPI Registers]. eSPI BAR={Spi_eSpi_BaseAddr[31:6], 000000b} + 0001_0000h. See 3.26.8.3 [eSPI Registers].
5:4	Reserved.
3	**RouteTpm2Spi**. Read-write. Reset: 0. 1=TPM cycles are routed to SPI bus with TPM_SPI_CS# asserted.
2	**AbortEnable**. Read-write. Reset: 0. LPC Abort enable.
1	**SpiRomEnable**. Read-write. Reset: 1. 1=SPI ROM is enabled if chip is strapped to SPI ROM. 0=SPI ROM is disabled.
0	**AltSpiCSEnable**. Read-write. Reset: 0. Alternative SPI CS enable.

D14F3xA4 EC_PortAddress

Bits	Description
31:16	Reserved.
15:1	**EcPortAddr15_1**. IF (D14F3xB8[EcPortHostAccessEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 17h. If this field is non-zero, [EcPortActive] is set to 1, and an IO cycle from host has address[15:1] == EcPortAddr15_1, the cycle is routed to IMC.
0	**EcPortActive**. Read-write. Reset: 1. 1=LPC can decode the address specified in [EcPortAddr15_1]. 0=LPC ignores it.

D14F3xB0 RomDmaSrcAddr

Bits	Description
31:6	**DmaStartAddr**. Read-write. Reset: 0. Specifies the starting DMA address to read from the ROM. Note this is not the same as the legacy DMA function. This is meant to be used by BIOS to fetch BOOT code quicker.
5:0	Reserved.

D14F3xB4 RomDmaDstAddr

Bits	Description
31:6	**DmaDstAddr**. Read-write. Reset: 0. Specifies the target DMA address to be written in the system memory. Note this is not the same as the legacy DMA function. This is meant to be used by BIOS to fetch BOOT code quicker.
5:0	Reserved.

D14F3xB8 RomDmaControl/EcControl/HostControl

Bits	Description
31:30	Reserved.
29	**PrefetchMissEnHost**. Read-write. Reset: 1. 1=Force the on-going prefetch to stop if the address of the pending read is not within the prefetch range for host.
28	**PrefetchArbNoSwitch**. Read-write. Reset: 1. 1=Only allow prefetch.
27	**TStartEnh**. Read-write. Reset: 0. BIOS: 1. 1=Enable hand-instance of the pulse generator; allow LPCCLK0/LPCCLK1 to stop under ClkRun# protocol.
26	**DisableSpiInSleep**. Read-write. Reset: 0. BIOS: 1. 1=SPI signals are driven to low and PwrGood goes to low. 0=SPI signals are left high and PwrGood goes to low.
25	**LpcBusPullUpEn**. Read-write. Reset: 0. BIOS: 1. 1=Leave pull-up for LFrame#, LDrq# to be enabled in S0 and disabled in S3/S5. 0=Force the pull-up for LFrame#, LDrq# to be disabled.
24	**PrefetchEnSpiFromHost**. Read-write. Reset: 0. BIOS: 1. This is for performance enhancement purpose. 1=SPI controller prefetches from the flash on behalf of the host.

23	**PrefetchEnSpiFromUSB**. Read-write. Reset: 0. BIOS: 1. This is for performance enhancement purpose. 1=SPI controller prefetches from the flash on behalf of USB3 controller.
22	**PrefetchEnSel**. Read-write. Reset: 1. 0=Prefetch only the address of the read request that is on the 4-byte boundary. 1=Prefetch only the address of the read request that is on the 8-byte boundary.
21	**PrefetchMissEn**. Read-write. Reset: 1. 1=Force the on-going prefetch to stop if the address of the pending read is not within the prefetch range for IMC.
20	**EcReadOfSwitch**. Read-write. Reset: 0. 1=Upper 16 bits of IMC read request address are specified in D14F3xC4. 0=Upper 16 bits of IMC read request address are specified by auto-rom detection logic, see D14F3xCC for auto-rom detection.
19	**EcPortHostAccessEn**. Read-write. Reset: 0. 1=Allow host to program D14F3xA4.
18	**PrefetchEnSpiFromEC**. Read-write. Reset: 0. 1=Enable prefetching a cache line (64 bytes) when IMC reads code from the SPI ROM. This bit can be programmed by IMC only.
17	**SpiHoldOnGevent9Dis**. Read-write. Reset: 0. 0=GEVENT9 and ROM_RST# are configured to SPI_HOLD# and SPI_WP# function. 1=GEVENT9 and ROM_RST# are not configured to SPI_HOLD# and SPI_WP# function.
16	**ECPageProtect**. Read-write. Reset: 0. Enable protection of the IMC page registers. 1=D14F3x[C7:BC] are only accessible by IMC; host writes have no effect and host reads return 0. 0=D14F3x[C7:BC] are accessible by both IMC and host.
15:6	**DWCount**. Read-write. Reset: 0. This register specifies the number of cache lines (64 bytes) to be fetched from the ROM when DMA is used.
5:3	Reserved.
2	**RomcpSupportAbRetry**. Read-write. Reset: 1. 0=ROM copy doesn't support AB retry. 1=ROM copy supports AB retry.
1	**DmaErrorStatus**. Read; Write-1-to-clear. Reset: 0. 1=Previous transfer has error. 0=Previous transfer has completed successfully.
0	**DmaStart**. Read; Write-1-only; Cleared-by-hardware. Reset: 0. Writing 1 to this bit causes LPC bridge starts the DMA function, with starting addresses defined by D14F3xB0 and D14F3xB4. This bit returns the status of the DMA transfer. A return value of 0 means the DMA transfer is complete. A return value of 1 means the DMA transfer is running.

D14F3xC0 EcRomWrOffset

Bits	Description
31:0	**EcRomWrOffset**. Read-write. Reset: FFF2_0000h. Specifies the address of IMC write requests. If D14F3xB8[ECPageProtect] == 1, this register returns its current value. If D14F3xB8[ECPageProtect] == 0, this register returns all 0s.

D14F3xC4 EcRomRdOffset

Bits	Description
31:0	**EcRomRdOffset**. Read-write. Reset: FFF2_0000h. Specifies the address of IMC read requests. If D14F3xB8[ECPageProtect] == 1, this register returns its current value. If D14F3xB8[ECPageProtect] == 0, this register returns all 0s.

D14F3xC8 ClientRomProtect

Bits	Description
31	**AutoSizeStart**. Read; Write-1-only. Reset: 0. IF (READ) THEN This bit returns the status of Auto-Rom detection. 1=The AutoRom detection is done. 0=AutoRom detection is not done yet. ELSE 1=Trigger AutoRom detection. 0=No effect. ENDIF.
30:3	Reserved.
2	**UsbRomProtectEn**. Read-write. Reset: 0. 1=Software cannot access USB portion of the flash.
1	Reserved.
0	**EcRomPortectEn**. Read-write. Reset: 0. 1=Software cannot access IMC portion of the flash.

D14F3xCC AutoRomCfg

Bits	Description
31:3	**AutoRomAddr**. Read-only. Reset: X. Based on the setting of AutoAddressSelect, this register returns the address value detected by the auto-rom detection logic (EcCodeOffset, XhciRomOffset).
2	**AutoSel**. Read-write. Reset: 0. 0=HW AutoSize function auto detects the EC, GEC, and USB3 firmware locations. 1=Software can override the EC, GEC, and USB3 firmware locations.
1:0	**AutoAddressSelect**. Read-write. Reset: 0. The FCH auto detects the IMC and USB3 firmware locations. After detection, it writes the address pointers of each firmware onto bits[31:3]. This field configures which address to be returned on bits[31:3]. Bits — Definition 00b — Reserved 01b — Reserved 10b — Reserved 11b — USB XHCI

D14F3xD0 ClkCntrl

Bits	Description
31	**ClkRunEn**. Read-write. Reset: 0. 1=ClkRun function is enabled and LPCCLK0/LPCCLK1 can be stopped. 0=ClkRun function is disabled and LPCCLK0/LPCCLK1 can be running all the time. Should be set to 1 for mobile platforms for energy savings.
30:24	**ClkRunDlyCounter**. Read-write. Reset: 8h. Specifies the amount of clocks to be extended before stopping the LPCCLK0/LPCCLK1.
23	Reserved.
22	**Lclk1ClkrunOvrid**. Read-write. Reset: 1. 1=LPCCLK1 is functioning with CLKRUN protocol. 0=LPCCLK1 is forced to run.
21	**Lclk0ClkrunOvrid**. Read-write. Reset: 1. 1=LPCCLK0 is functioning with CLKRUN protocol. 0=LPCCLK0 is forced to run.
20:15	Reserved.
14	**Lclk1En**. Read-write. Reset: 1. 0=LPCCLK1 is forced to stop. 1=LPCCLK1 is functioning with CLKRUN protocol.

13	**Lclk0En**. Read-write. Reset: 1. 0=LPCCLK0 is forced to stop. 1=LPCCLK0 is functioning with CLKRUN protocol.
12:3	Reserved.
2	**SpiOnClkRun**. Read-write. Reset: 1. 0=Spi request can assert ClkRun#. 1=Spi request doesn't assert ClkRun#.
1:0	**ClkGateCntrl**. Read-write. Reset: 2h. These two bits control whether the LPC module allows clock gating to the internal 66MHz core clock. Bits — Definition 00b — Disable the clock gating function. 01b — Wait 16 clocks before allowing clock gating to the LPC module. 10b — Wait 64 clocks before allowing clock gating to the LPC module. 11b — Wait 256 clocks before allowing clock gating to the LPC module.

D14F3xD4 ClkRunOption

Bits	Description
31:8	Reserved.
7:4	**MinAssertion**. Read-write. Reset: 4h. Specifies the minimum time of ClkRun# assertion. The unit is 30 ns.
3:2	Reserved.
1	**ExtendClkRunDrv.** Read-write. Reset: 1. BIOS: 0. Controls the driving of ClkRun# if the host has no request pending. 0=Drive ClkRun# to H (30ns) before releasing if there is no request. 1=Release ClkRun# if there is no request.
0	**ClkRunInputDly**. Read-write. Reset: 0. 0=Delay 30 ns before capturing ClkRun# input. 1=Delay 60 ns before capturing ClkRun# input.

3.26.8.2 SPI Registers

SPI configuration registers are accessed through SPI base address specified by D14F3xA0. Software can communicate with the SPI ROM through the default memory or alternate program method:

- Memory access to the BIOS ROM address space is automatically handled by the hardware. The SPI ROM controller translates the memory address onto the SPI bus and accesses the SPI ROM data. Any other commands besides memory read or memory write to the SPI ROM need to go through the alternate program method.
- In alternate program method, software needs to program the SpiOpCode, SpiAddress, TxByteCount, RxByteCount, put the data into the transmit FIFO, and then execute the command. The hardware communicates with the SPI ROM using these parameters. This alternate method basically allows software to issue any flash vendor specific commands such as ERASE and STATUS.

SPIx00 SPI_Cntrl0

Bits	Description
31	**SpiBusy**. Read-only; Updated-by-hardware. Reset: 0. 0=SPI bus is idle. 1=SPI bus is busy.

30:29	**SpiReadMode[2:1]**. Read-write. Reset: 0. This field, along with SpiReadMode[0] below, combine to specify the SPI read mode: Bits — Definition 000b — Normal read (up to 33M) 001b — Reserved 010b — Dual IO (1-1-2) 011b — Quad IO (1-1-4) 100b — Dual IO (1-2-2) 101b — Quad IO (1-4-4) 110b — Normal read (up to 66M) 111b — Fast Read
28	**SpiClkGate**. Read-write. Reset: 0. 1=Skip the 8th SPI clock at the end data when doing read.
27	Reserved.
26:24	**ArbWaitCount**. Read-write. Reset: 7h. Specifies the amount of wait time the SPI controller asserts HOLD# before it should access the SPI ROM, under ROM sharing mode with the MAC.
23	**SpiHostAccessRomEn**. Read; Write-0-only. Reset: 1. This is a clear-once protection bit; once it is cleared to 0 it cannot be set back to 1. 0=MAC cannot access BIOS ROM space (upper 512KB). 1=MAC can access BIOS ROM space.
22	**SpiAccessMacRomEn**. IF (Mode == SMI) Read-write. ELSE Read; Write-0-only. ENDIF. Reset: 1. This is a clear-once protection bit. 0=Software cannot access MAC's portion of the ROM space (lower 512KB). 1=Software can access MAC's portion of the ROM space.
21	**IllegalAccess**. Read-only; Updated-by-hardware. Reset: 0. 0=Legal index mode access. 1=Illegal index mode access.
20	**FifoPtrClr**. RAZ; Write-1-only. Reset: 0. Writing 1 to this bit clears the internal FIFO pointer at SPIx0C[FifoPtr].
19	**SpiArbEnable**. Read-write. Reset: 0. If a MAC is sharing the ROM with the FCH, both chips need to go through an arbitration process before either one can access the ROM. If MAC is not sharing the SPI ROM, BIOS should set this bit to 0 to speed up the SPI ROM access. 1=Enable the arbitration.
18	**SpiReadMode[0]**. Read-write. Reset: 0. Bit 0 of SpiReadMode. See the definition of SpiRead-Mode[2:1] in this register.
17	Reserved.
16	**ExecuteOpCode**. Write-1-only; Cleared-by-hardware. Reset: 0. Write 1 to execute the transaction in the alternate program registers. This bit returns to 0 when the transaction is complete. If the command is an illegal command, the bit cannot be set and thereby cannot execute.
15:8	Reserved.
7:0	**SpiOpCode**. Read-write. Reset: 0. Specifies the SPI opcode in alternate program method.

SPIx04 SPI_RestrictedCmd

Bits	Description
31:24	**RestrictedCmd3**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as RestrictedCmd0.
23:16	**RestrictedCmd2**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as RestrictedCmd0.

15:8	**RestrictedCmd1**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as RestrictedCmd0.
7:0	**RestrictedCmd0**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0.

SPIx08 SPI_RestrictedCmd2

Bits	Description
31:24	**RestrictedCmdWoAddr2**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as [RestrictedCmdWoAddr0].
23:16	**RestrictedCmdWoAddr1**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as [RestrictedCmdWoAddr0].
15:8	**RestrictedCmdWoAddr0**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as SPIx04[RestrictedCmd0] except that this field defines a restricted command that does not have address.
7:0	**RestrictedCmd4**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0. Same as SPIx04[RestrictedCmd0].

SPIx0C SPI_Cntrl1

Bits	Description
31:24	**ByteCommand**. Read-write. Reset: 2h. Specifies the command byte for the opcode transaction.
23:22	Reserved.
21:16	**WaitCount**. Read-write. Reset: 22h. Specifies the time unit = 15 ns * (WaitCount+1).
15:12	Reserved.
11	**TrackMacLockEn**. Read-write. Reset: 0. 1=The SPI controller locks the SPI from the MAC when it has detected a command from the MAC matching the value defined in SPIx10[MacLockCmd0] or SPIx10[MacLockCmd1]. 0=The SPI controller unlocks the bus when it has detected a command from the MAC matching the value defined in SPIx10[MacUnlockCmd0] or SPIx10[MacUnlockCmd1].
10:8	**FifoPtr**. Read-only. Reset: 0. This specifies the internal pointer location, which can be cleared through SPIx00[FifoPtrClr].
7:0	**SpiParameters**. Read-write. Reset: 0. This is the TX/RX FIFO port which can take up to 8 bytes. To send data to SPI ROM, software writes data into this port. To retrieve data that are received from the SPI ROM, software reads from this port.

SPIx10 SPI_CmdValue0

Bits	Description
31:24	**MacUnlockCmd1**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 4h. Same as MacUnlockCmd0.

23:16	**MacUnlockCmd0**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 4h. This field is used to compare against the opcode sent out by the MAC. If SPIx0C[TrackMacLockEn] == 1, the controller unlocks the SPI bus for the MAC. In other words, access by the CPU is allowed again.
15:8	**MacLockCmd1**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 20h. Same as MacLockCmd0.
7:0	**MacLockCmd0**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 6h. This is used to compare against the opcode sent out by the MAC. If SPIx0C[TrackMacLockEn] == 1, the controller locks the SPI bus for the MAC. In other words, the MAC has the exclusive access to the ROM and access by the CPU is delayed until this is unlocked. This allows the MAC to do a certain sequence of operations without interruption.

SPIx14 SPI_CmdValue1

Bits	Description
31:24	**RDSR**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 5h. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the RDSR (read status register) command from the MAC.
23:16	**RDID**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 9Fh. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the RDID (read ID) command from the MAC.
15:8	**WRDI**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 4h. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the WRDI (write disable) command from the MAC.
7:0	**WREN**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 6h. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the WREN (write enable) command from the MAC.

SPIx18 SPI_CmdValue2

Bits	Description
31:24	**BYTEWR**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 2h. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the BYTEWR (byte write) command from the MAC.
23:16	**PAGEWR**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 0Ah. Page write command.

15:8	**FRead**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: Bh. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the FRead (fast read) command from the MAC.
7:0	**Read**. IF (SPIx00[SpiAccessMacRomEn] == 0 \|\| SPIx00[SpiHostAccessRomEn] == 0) THEN Read-only. ELSE Read-write. ENDIF. Reset: 3h. This is used to compare against the opcode sent out by the MAC. This is a predefined value to decode for the Read (read byte) command from the MAC.

SPIx1C Reserved

Bits	Description
7:0	Reserved.

SPIx1D Alt_SPI_CS

Bits	Description
7	**SpiCsDlySel**. Read-write. Reset: 1. 1=125 ns minimum SPI_CS# de-assertion time. 0=75 ns minimum SPI_CS# de-assertion time.
6:3	Reserved.
2	**WriteBufferEn**. Read-write. Reset: 0. SPI write performance enhancement. 1=SPI bridge can take burst write from the host and transfer it to the SPI flash.
1:0	**AltSpiCsEn**. Read-write. Reset: 0. These two bits enable the alternate SPI_CS#. Bits — Definition 00b — select xSPI_CS# 01b — select xSPI_CS1# 10b — select xSPI_CS2# 11b — select xSPI_CS3#

SPIx1E SpiExtRegIndx

Bits	Description
7:0	**SPI_ExtReg_Indx**. Read-write. Reset: 0. Specifies the offset of the extended SPI register to be read/written from SPIx1F.

SPIx1F SpiExtRegData

Bits	Description
7:0	**SPI_ExtReg_Data**. Read-write. Reset: 3Bh. Specifies the read data or write data of the extended SPI register.

SPIx1F_x00 DDR_CMD

Bits	Description
7:0	**DDR_CMD**. Read-write. Reset: 3Bh. Specifies the command value to be used when host is doing a Dual Output Read.

SPIx1F_x01 QDR_CMD

Bits	Description
7:0	**QDR_CMD**. Read-write. Reset: 6Bh. Specifies the command value to be used when host is doing a Quad Output Read.

SPIx1F_x02 DPR_CMD

Bits	Description
7:0	**DPR_CMD**. Read-write. Reset: BBh. Specifies the command value to be used when host is doing a Dual IO High Performance Read.

SPIx1F_x03 QPR_CMD

Bits	Description
7:0	**QPR_CMD.** Read-write. Reset: EBh. Specifies the command value to be used when host is doing a Quad IO High Performance Read.

SPIx1F_x04 ModeByte

Bits	Description
7:0	**ModeByte**. Read-write. Reset: 0.

SPIx1F_x05 TxByteCount

Bits	Description
7:0	**TxByteCount**. Read-write. Reset: 0. Specifies the number of bytes to be sent to SPI ROM.

SPIx1F_x06 RxByteCount

Bits	Description
7:0	**RxByteCount**. Read-write. Reset: 0. Specifies the number of bytes to be received from the SPI ROM.

SPIx1F_x07 SPIDataFifoPtr

Bits	Description
7:0	**SpiDataFiFoPtr**. Read-only. Reset: 0. Specifies the current pointer of read/write Data FIFO.

SPIx20 SPI100 Enable

Bits	Description
7:1	Reserved.
0	**UseSpi100**. Read-write. Reset: 0. BIOS:1. 0=Does not support 100 MHz speed. 1=Supports 100 MHz speed. The actual read speed also depends on SPIx22.

SPIx22 SPI100 Speed Config

Bits	Description
15:12	**NormSpeed[3:0]**. Read-write. Reset: 0011b. Configures the SPI bus normal speed in SPI100 engine. If the command is not using TpmSpeed and FastSpeed, it will use NormSpeed. Bits / Definition / Bits / Definition 0000b / 66.66 MHz / 0100b / 100 MHz 0001b / 33.33 MHz / 0101b / 800 KHz 0010b / 22.22 MHz / 1111b-0110b / Reserved 0011b / 16.66 MHz
11:8	**FastSpeed[3:0]**. Read-write. Reset: 0001b. Configures the SPI bus speed for the following command in SPI100 engine: • FAST READ • DDR READ (1-1-2) • QDR READ (1-1-4) • DPR READ (1-2-2) • QPR READ (1-4-4) Bits / Definition / Bits / Definition 0000b / 66.66 MHz / 0100b / 100 MHz 0001b / 33.33 MHz / 0101b / 800 KHz 0010b / 22.22 MHz / 1111b-0110b / Reserved 0011b / 16.66 MHz

7:4	**AltSpeed[3:0]**. Read-write. Reset: 0011b. Configures the SPI bus speed for the AltOpCode mode in SPI100 engine. Bits / Definition / Bits / Definition 0000b 66.66 MHz 0100b 100 MHz 0001b 33.33 MHz 0101b 800 KHz 0010b 22.22 MHz 1111b-0110b Reserved 0011b 16.66 MHz
3:0	**TpmSpeed[3:0]**. Read-write. Reset: 0011b. Configures the SPI bus speed for TPM read and write in SPI100 engine. Bits / Definition / Bits / Definition 0000b 66.66 MHz 0100b 100 MHz 0001b 33.33 MHz 0101b 800 KHz 0010b 22.22 MHz 1111b-0110b Reserved 0011b 16.66 MHz

SPIx2C SPI100 Host Prefetch Config

Whenever there is read from host, the SPI starts to prefetch ROM data into the Host Prefetch Buffer. The prefetch starts from the first address requested by host, and finish when any of the following happens:

- We have reached the maximum prefetch size defined in HostPrefetchSize and HostPrefOn64ByteBoundary register.
- When host requests an address that is not already prefetched and not going to be prefetched shortly, the SPI stops current prefetch action and re-starts a new prefetch with the first address being the current address requested by the Host.
- When there is a ROM-Write or AltOpCode request from host, the on-going prefetch is terminated, and the prefetch buffer is flushed.
- When there is a TPM-Write, TPM-Read, USB-Read or EC-Read request, the on-going prefetch is halted. The contents of prefetch buffer will be preserved so that the host can access them later.

There are two prefetch prediction algorithms selected by HostWillHitEn and HostHitSoonEn:

- "Will Hit" algorithm. The current requested address will be fetched "shortly" if this equation is true: Current Requested Address <= (First Prefetched Address + HostPrefetchSize).
- "Hit Soon" algorithm. The current requested address will be fetched "shortly" if this equation is true: Current Requested Address <= (Last Prefetched Address + HostHitRange).

Of course, the current requested address also has to be equal to or larger than the first prefetched address. If current requested address meet the criteria, the on-going prefetch continues and host waits until the prefetch buffer receives the requested data.

Bits	Description
15	**Rd4dw_en_host**. Read-write. Reset: 1. BIOS:0. 1=Enable host burst to 4 Dwords.
14	**HostBurstEn**. Read-write. Reset: 1. 1=Enable host buffer burst data out.
13	**HostHitSoonEn**. Read-write. Reset: 0. 1=Enable the "hit soon" algorithm.
12	**HostWillHitEn**. Read-write. Reset: 1. 1=Enable the "will hit" algorithm. When "will hit" algorithm is enabled, "hit soon" algorithm is automatically disabled regardless of the setting in HostHitSoonEn.
11:8	**HostHitRange[3:0]**. Read-write. Reset: 0100b. Configures the "hit range" in the "hit soon" algorithm.

7	**HostPrefOn64ByteBoundary**. Read-write. Reset: 1. 0=Always fetch 64 bytes no matter whether the first Host requested address lies on the 64-byte boundary or not. 1=Fetch 64 bytes if the first Host requested address lies on the 64-byte boundary; Otherwise, just get the 4 bytes Host is currently requesting.
6:0	**HostPrefetchSize[6:0]**. Read-write. Reset: 100_0000b. Configures the maximum prefetch byte count for the host prefetch buffer. The value has to be less than or equal to 64.

SPIx40 DDRCmdCode

Bits	Description
7:0	**DDR_CMD**. Read-write. Reset: 3Bh. This is the shadow register of SPIx1F_x00.

SPIx41 QDRCmdCode

Bits	Description
7:0	**QDR_CMD**. Read-write. Reset: 6Bh. This is the shadow register of SPIx1F_x01.

SPIx42 DPRCmdCode

Bits	Description
7:0	**DPR_CMD**. Read-write. Reset: BBh. This is the shadow register of SPIx1F_x02.

SPIx43 QPRCmdCode

Bits	Description
7:0	**QPR_CMD**. Read-write. Reset: EBh. This is the shadow register of SPIx1F_x03.

SPIx44 ModeByte

Bits	Description
7:0	**ModeByte**. Read-write. Reset: 0. This is the shadow register of SPIx1F_x04.

SPIx45 CmdCode

Bits	Description
7:0	**SpiOpCode**. Read-write. Reset: 0. This is the shadow register of SPIx00[SpiOpCode].

SPIx47 CmdTrigger

Bits	Description
7	**Excute**. Write-1-only; Cleared-by-hardware. Reset: 0. This bit is the shadow register of SPIx00[ExecuteOpCode].
6:0	Reserved.

SPIx48 TxByteCount

Bits	Description
7:0	**TxByteCount**. Read-write. Reset: 0. This is the shadow register of SPIx1F_x05.

SPIx4B RxByteCount

Bits	Description
7:0	**RxByteCount**. Read-write. Reset: 0. This is the shadow register of SPIx1F_x06.

SPIx4C SpiStatus

Bits	Description
31	**SpiBusy**. Read-only; Updated-by-hardware. Reset: 0. 1=SPI bus is busy. 0=SPI bus is idle.
30:23	Reserved.
22:16	**FiFoRdPtr**. Read-only; Updated-by-hardware. Reset: 0. The current Data FIFO read pointer.
15	Reserved.
14:8	**FiFoWrPtr**. Read-only; Updated-by-hardware. Reset: 0. The current Data FIFO write pointer.
7:0	**DoneByteCount**. Read-only; Updated-by-hardware. Reset: 0. Indicates how many bytes has been received or sent in the previous SPI transaction.

SPIx[C6:80] FIFO[70:0]

Bits	Description
7:0	**FiFo**. Read-write. Reset: 0. Contains the Data FIFO byte which is used in command mode to send or receive data.

3.26.8.3 eSPI Registers

eSPI (enhanced Serial Peripheral Interface) configuration registers are accessed through MMIO base address specified by {D14F3xA0[Spi_eSpi_BaseAddr[31:6]], 000000b} + 0001_0000h.

eSPIx00 eSPI Software Specific Register 0

Bits	Description
31:24	**VWDataOOBPktByte2**. Read-write. Reset: 0. Channel independent command selected: • 00h. Pchannel selected: • Length[7:0]. VW channel selected: • Data for VW group 0. OOB channel selected: • Length[7:0].
23:16	**AddrByte0VWIdxOOBPktByte1**. Read-write. Reset: 0. Channel independent command selected: • Address[7:0] of SET_CONFIGURATION and GET_CONFIGURATION command. • [15:12]: 0h. • [11:8]: address[11:8]. Note: these bits are ignored in in-band command. P channel selected: • [15:12]: Tag • [11:8]: Length[11:8]. VW channel selected: • Index for VW group 0. OOB channel selected: • [15:12]: Tag • [11:8]: Length[11:8].
15:8	**AddrByte0PCycTypeVWCntOOBCycType**. Read-write. Reset: 0 Address or P Cycle Type for OOB message or VWcount. Channel independent command selected: • Address[15:8] of SET_CONFIGURATION and GET_CONFIGURATION. • Bit[1:0] need to be 0. Note: These bits are ignored in in-band command. P channel selected: • Message cycle type (0001xxxy) to instruct eSPI controller to send peripheral message with data (8 bytes+data byte N) or message without data (8 bytes). VW channel selected: • Bit[5:0] represent how many VWgroups to be communicated in the same packet. Note: In the current design, it is limited to 16 VW groups with bit[5:4] set to 0. OOB channel selected: • Program this byte to be 21h to instruct the eSPI controller to send tunneled SMBUS message to slave.
7:6	Reserved.
5:4	**SlaveNSelect.** Read-write. Reset: 0. Slave N selected: Bits — Definition 00b — Slave 0 selected. 01b — Reserved. 10b — Reserved. 11b — Reserved.

3	**CommandStatus**. Read-write. Reset: 0. 1=This bit needs to be set after software has configured all eSPI specific registers to inform the protocol layer to send the command or packet. 0=Hardware reset this bit after the packet is sent down.
2:0	**SWCommandType**. Read-write. Reset: 0. Software command to send down to slave. Bit[2] = 0 selects a channel independent command. Bit[2] = 1 selects a P/VW/OOB packet to send down. Bits — Definition 000b — SET_CONFIGURATION. 001b — GET CONFIGURATION. 010b — In-band RESET command. 011b — Reserved. 100b — P (Peripheral) channel message down stream. 101b — VW (Virtual Wire) channel down stream. 110b — OOB (Out of Band) channel down stream. 111b — Reserved.

eSPIx04 eSPI Software Specific Register 1

Bits	Description
31:24	**VWDataOOBPktByte6**. Read-write. Reset: 0. Channel independent command selected: • Data[31:24]. P channel selected: • Message specific byte 2. VW channel selected: • Data for VW group 2 if VW count is greater than or equal to 2. OOB channel selected: • SMBus data byte 0.
23:16	**VWIdxOOBPktByte5**. Read-write. Reset: 0. Channel independent command selected: • Data[23:16]. P channel selected: • Message specific byte 1. VW channel selected: • Index for VW group 2 if VW count is greater than or equal to 2. OOB channel selected: • SMBus byte count which needs to be programmed not greater than 32 bytes as SMBus specification.

15:8	**VWDataOOBByteCnt**. Read-write. Reset: 0 Channel independent command selected: • Data[15:8]. P channel selected: • Message specific byte 0. VW channel selected: • Data for VW group 1 if VW count is greater than or equal to 1. OOB channel selected: • SMBus command operation code.
7:0	**VWIdxOOBPktbyte3**. Read-write. Reset: 0. Channel independent command selected: • Data[7:0]. P channel selected: • Message-code[7:0]. VW channel selected: • Index for VW group 1 if VW count is greater than or equal to 1. OOB channel selected: • SMBus slave address. Bit[0] needs to be programmed to '1'.

eSPIx[1C:08:Step6] eSPI Software Specific Register 2-7

Bits	Description
31:24	**PDataByte2VWDataOOBPktByte4**. Read-write. Reset: 0. P channel selected: • Data byte 2, 6, 10, 14, 18, 22 when offset [08], [0C], [10], [14], [18], [1C] respectively. VW channel selected: • Data for VW group 5, 7, 9, 11, 13, 15 if VW count is greater than or equal to 5, 7, 9, 11, 13, 15 respectively. OOB channel selected: • SMBus data byte 4, 8, 12, 16, 20, 24 if OOB byte count is greater than or equal to 4, 8, 12, 16, 20, 24 respectively.
23:16	**PDataByte1VWIdxOOBPktByte3**. Read-write. Reset: 0. P channel selected: • Data byte 1, 5, 9, 13, 17, 21 when offset [08], [0C], [10], [14], [18], [1C] respectively. VW channel selected: • Data for VW group 4, 6, 8, 10, 12, 14 if VW count is greater than or equal to 4, 6, 8, 10, 12, 14 respectively. OOB channel selected: • SMBus data byte 3, 7, 11, 15 19, 23 if OOB byte count is greater than or equal to 3, 7, 11, 15, 19, 23 respectively.

15:8	**PDataByte0VWDataOOBPktByte2**. Read-write. Reset: 0. P channel selected: • Data byte 0, 4, 8, 12, 16, 20 when offset [08], [0C], [10], [14], [18], [1C] respectively. VW channel selected: • Data for VW group 3, 5, 7, 9, 11, 13 if VW count is greater than or equal to 3, 5, 7, 9, 11, 13 respectively. OOB channel selected: • SMBus data byte 2, 6, 10, 14 18, 22 if OOB byte count is greater than or equal to 2, 6, 10, 14, 18, 22 respectively.
7:0	**PMsgByte3DataByte3VWIdxOOBPktByte1**. Read-write. Reset: 0. P channel selected: • Message specific byte 3 when offset [08]. • Data byte 3, 7, 11, 15, 19 when offset [0C], [10], [14], [18], [1C] respectively. VW channel selected: • Index for VW group 3, 5, 7, 9, 11, 13 if VW count is greater than or equal to 3, 5, 7, 9, 11, 13 respectively. OOB channel selected: • SMBus data byte 1, 5, 9, 13, 17, 21 if OOB byte count is greater than or equal to 1, 5, 9, 13, 17, 21 respectively.

eSPIx20 eSPI Software Specific Register 8

Bits	Description
31:24	**PDataByte26OOBPktByte28**. Read-write. Reset: 0. P channel selected: • Data byte 26 if data byte is greater than or equal to 26. VW channel selected: • Not applicable. OOB channel selected: • SMBus data byte 28 if OOB byte count is greater than or equal to 28.
23:16	**PDataByte25OOBPktByte27**. Read-write. Reset: 0. P channel selected: • Data byte 25 if data byte is greater than or equal to 25. VW channel selected: • Not applicable. OOB channel selected: • SMBus data byte 27 if OOB byte count is greater than or equal to 27.

15:8	**PDataByte24VWDataOOBPktByte26**. Read-write. Reset: 0. P channel selected: • Data byte 24 if data byte is greater than or equal to 24. VW channel selected: • Data for VW group 15 if VW count is greater than or equal to 15. OOB channel selected: • SMBus data byte 26 if OOB byte count is greater than or equal to 26.
7:0	**PDataByte23VWIdxOOBPktByte25**. Read-write. Reset: 0. P channel selected: • Data byte 23 if data byte is greater than or equal to 23. VW channel selected: • Index for VW group 15 if VW count is greater than or equal to 15. OOB channel selected: • SMBus data byte 25 if OOB byte count is greater than or equal to 25.

eSPIx24 eSPI Software Specific Register 9

Bits	Description
31:24	**PDataByte30OOBPEC**. Read-write. Reset: 0. P channel selected: • Data byte 30 if data byte is greater than or equal to 30. VW channel selected: • Invalid. OOB channel selected: • PEC.
23:16	**PDataByte29OOBPktByte31**. Read-write. Reset: 0. P channel selected: • Data byte 29 if data byte is greater than or equal to 29. VW channel selected: • Invalid. OOB channel selected: • SMBus data byte 31 if OOB byte count is greater than or equal to 31.
15:8	**PDataByte28OOBPktByte30**. Read-write. Reset: 0. P channel selected: • Data byte 28 if data byte is greater than or equal to 28. VW channel selected: • Invalid. OOB channel selected: • SMBus data byte 30 if OOB byte count is greater than or equal to 30.
7:0	**PDataByte27OOBPktByte29**. Read-write. Reset: 0. P channel selected: • Data byte 27 if data byte is greater than or equal to 27. VW channel selected: • Invalid. OOB channel selected: • SMBus data byte 29 if OOB byte count is greater than or equal to 29.

eSPIx28 eSPI Software Specific Register 10

Bits	Description
31:8	Reserved.
7:0	**PDataByte31**. Read-write. Reset: 0. P channel selected: • Data byte 31 if data byte is greater than or equal to 31.

eSPIx2C eSPI Master Capability

Bits	Description
31	**CRCCheck**. Read-only. Reset: 1. 0=Master support CRC checking. 1=Master doesn't support CRC checking.
30	**AlertMode**. Read-only. Reset: 1. 0=IO pin_1 is used to signal the Alert event. 1=A dedicated Alert# pin is used to signal the Alert event, or IO pin_1 used for Alert.
29:28	**IOMode**.Read-only. Reset: 2h. Encoding Support Operating Mod. Bits Definition 00b Single IO 01b Dual IO, Single IO 10b Quad IO, Dual IO, Single IO 11b Reserved
27:25	**OperatingSupportFreq**. Read-only. Reset: 3h. Bits Definition 000b 16.7 Mhz 001b 16.7 Mhz, 33 Mhz 011b 16.7 Mhz, 33 Mhz, 66 Mhz
24:22	**NumberOfSlave**. Read-only. Reset: 1. Indicating the number of slaves. Bits Definition 111b-001b <NumberOfSlave> slaves 000b 8 slaves
21:19	**PChannelMaxPayloadSize**. Read-only. Reset: 1. The payload of the transaction must not cross the naturally aligned address boundary of the corresponding Maximum Payload Size. Bits Definition 000b Reserved. 001b 64 bytes max payload size. 111b-010b Reserved.
18:13	**OperatingMaxVWCount**. Read-only. Reset: 0Fh. The maximum number of Virtual Wire groups that can be sent in a single Virtual Wire packet. This is a 0-based count. The default value of 0 indicates count of 1. The value configured in this field must never be more than the value advertised in the Maximum Virtual Wire Count Supported field.
12:10	**OOBMessageChannelMaxPayload**. Read-only. Reset: 1. This field advertises the Maximum Payload Size supported by the Master. Bits Definition 000b Reserved. 001b 64 bytes max payload size. 111b-010b Reserved.

9:7	**FlashAccessChannelMaxPayload**. Read-only. Reset: 1. Bits Definition 000b Reserved. 001b 64 bytes max payload size. 111b-010b Reserved.
6:4	**MasterVersion**. Read-only. Reset: 0. 0=Master is 0.7 version. 1=Master is 0.75 version.
3	**PChannelSupport**. Read-only. Reset: 1. 1=Supported. 0=Not supported.
2	**VWChannelSupport**. Read-only. Reset: 1. 1=Supported. 0=Not supported.
1	**OOBMessageChannelSupport**. Read-only. Reset: 1. 1=Supported. 0=Not supported.
0	**FlashAccessChannelSupport**. Read-only. Reset: 0. 1=Supported. 0=Not supported.

eSPIx30 eSPI Global Control and Status Register 0

Bits	Description
31:30	Reserved.
29:24	**WaitStateCounter**. Read-write. Reset: 0. Specifies the timeout count for wait state.
23:8	**WatchDogCounter**. Read-write. Reset: 0. Specifies the timeout retry count for PCI downstream retries.
7	Reserved.
6:4	**Glb_Alink_Idle_Timer:Global Alink Idle Timer**. Read-write. Reset: 0. Set the bits to select different Idle timer timeout value. Once the Idle timer reach the timeout value and “Global Alink clock gating Enable” set, eSPI outputs ESPI_Stop_AlClk to do global Alink clock gating. Bits Definition 000b 16 clocks 001b 32 clocks 010b 64 clocks 011b 128 clocks 100b 256 clocks 101b 512 clocks 110b 1024 clocks 111b 2048 clocks
3	**Glb_Alink_CLK_gating_En:Global Alink clock gating Enable**. Read-write. Reset: 1. Set the bit to enable eSPI generating ESPI_Stop_AlClk to do global Alink clock gating once “Global Alink Idle Timer” reach the timeout value.
2	**Periphreal_clk_gating_En.** Read-write. Reset: 0. Peripheral clock gating Enable. Set the bit to enable Peripheral block do dynamic clock gating once the Slave Peripheral channel is disabled.
1	**WaitStateControlEnable**. Read-write. Reset: 0. Set the bit to enable the Wait State counter during eSPI bus turn around.
0	**WatchdogEnable**. Read-write. Reset: 0. Set the bit to enable the watchdog counter for all the PCI downstream transactions for eSPI.

eSPIx34 eSPI Global Control and Status Register 1

Bits	Description
31:22	Reserved.
21	**PBlkProtectEn: Peripheral Block Protection Enable**. Read-write. Reset: 0. 1=Peripheral Block checks CRC ERR and provides protection support. 0=Register Block only reports CRC ERR and does not provide any protection support.
20	**RegBlkProtectEn: Register Block Protection Enable**. Read; Write-1-to-clear. Reset: 0. 1=Register Block checks CRC ERR and provides protection support. 0=Register Block only reports CRC ERR and does not provide any protection support.
19:18	Reserved.
17:13	**RegisterCMDInterruptMapping**. Read-write. Reset: 17h. When a register command (downstream/upstream peripheral message, downstream/upstream OOB, downstream VW, channel independent command) has completed, the eSPI controller generates an interrupt to an interrupt pin according to the following field setting: Bits Definition 0_0000b IRQ0 0_0001b IRO1 0_0010b IRQ2 0_0011b IRQ3 0_0100b IRQ4 0_0101b IRO5 0_0110b IRQ6 0_0111b IRQ7 0_1000b IRQ8 0_1001b IRO9 0_1010b IRQ10 0_1011b IRQ11 0_1100b IRQ12 0_1101b IRO13 0_1110b IRQ14 0_1111b IRQ15 1_0000b IRQ16 1_0001b IRO17 1_0010b IRQ18 1_0011b IRQ19 1_0100b IRQ20 1_0101b IRO21 1_0110b IRQ22 1_0111b IRQ23 1_1110b-1_1000b Reserved 1_1111b SMI#

12:8	**ErrorInterruptMapping**. Read-write. Reset: 1Fh. When a slave transaction has error occurs, and the error interrupt enable is set, an error interrupt is generated to an interrupt pin according to the following field setting: Bits — Definition 0_0000b — IRQ0 0_0001b — IRO1 0_0010b — IRQ2 0_0011b — IRQ3 0_0100b — IRQ4 0_0101b — IRO5 0_0110b — IRQ6 0_0111b — IRQ7 0_1000b — IRQ8 0_1001b — IRO9 0_1010b — IRQ10 0_1011b — IRQ11 0_1100b — IRQ12 0_1101b — IRO13 0_1110b — IRQ14 0_1111b — IRQ15 1_0000b — IRQ16 1_0001b — IRO17 1_0010b — IRQ18 1_0011b — IRQ19 1_0100b — IRQ20 1_0101b — IRO21 1_0110b — IRQ22 1_0111b — IRQ23 1_1110b-1_1000b — Reserved 1_1111b — SMI#
7:5	Reserved.
4:3	S**ubtractiveDecodeSlaveSelection**. Read-write. Reset: 0. Bits — Definition 00b — Slave0 01b — Slave1 10b — Slave2 11b — Slave3
2	**SubtractiveDecodeEnable**. Read-write. Reset: 0. Enable eSPI to do Subtractive Decode.
1	**BusMasterEnable**. Read-write. Reset: 0. Enable eSPI Upstream Memory cycle posting.
0	**ControllerReset**. Read-write. Reset: 0. Set the bit to do global controller resets for eSPI controller, all the state machines return to idle, and all the request is flushed. All the configuration registers reset to the default value and software must send an In-Band Reset to each Slave device after the controller resets so that both Master and Slave run in same configuration mode.

eSPIx38 eSPI MISC Control Register 0

Bits	Description
31	**DebugUpDMA0Length**. Read-write. Reset: 0. Enable eSPI controller supported error reporting of upstream DMA read-write and downstream reads with 0 lengths. 1=Disable reporting.
30:15	Reserved.
14	**DebugbusAsyncFiFOPatternEN: Debug Bus Async FIFO Sequential Pattern Enable**. Read-write. Reset: 0.
13:12	**Debugbus_Sel_Slave: eSPI Slave Selection for Debug**. Read-write. Reset: 0. Bits Definition 00b Slave0 01b Slave1 10b Slave2 11b Slave3
11:6	**DebugbusMidGrp: Debug Bus Middle group**. Read-write. Reset: 31h.
5:0	**DebugbushighGrp: Debug Bus High group**. Read-write. Reset: 30h.

eSPIx3C eSPI MISC Control Register 1

Bits	Description
31:0	Reserved.

eSPIx40 eSPI IO or MMIO Decoding Enable Register for Slave N

Bits	Description
31:16	Reserved.
15	**ProgMMIORange3DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
14	**ProgMMIORange2DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
13	**ProgMMIORange1DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
12	**ProgMMIORange0DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
11	**ProgIORange3DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
10	**ProgIORange2DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
9	**ProgIORange1DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
8	**ProgIORange0DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
7:2	Reserved.
1	**IORange6064DecodeEnable**. Read-write. Reset: 0. 1=Claim this range.
0	**IORange2E2FDecodeEnable**. Read-write. Reset: 0. 1=Claim this range.

eSPIx44 eSPI IO Target Range Register 0 for Slave N

Bits	Description
31:16	**ProgIORange1BaseAddress**. Read-write. Reset: 0.
15:0	**ProgIORange0BaseAddress**. Read-write. Reset: 0.

eSPIx48 eSPI IO Target Range Register 1 for Slave N

Bits	Description
31:16	**ProgIORange3BaseAddress**. Read-write. Reset: 0.
15:0	**ProgIORange2BaseAddress**. Read-write. Reset: 0.

eSPIx4C eSPI IO Target Range Register 2 for Slave N

Bits	Description
31:24	**ProgIORange3Size**. Read-write. Reset: 0. 0=1 byte, FFh=256 bytes.
23:16	**ProgIORange2Size**. Read-write. Reset: 0. 0=1 byte, FFh=256 bytes.
15:8	**ProgIORange1Size**. Read-write. Reset: 0. 0=1 byte, FFh=256 bytes.
7:0	**ProgIORange0Size**. Read-write. Reset: 0. 0=1 byte, FFh=256 bytes.

eSPIx50 eSPI MMIOTarget Range Register 0 for Slave N

Bits	Description
31:0	**ProgMMIORange0Addr**. Read-write. Reset: 0. MMIO decode base address for Range 0.

eSPIx54 eSPI MMIOTarget Range Register 1 for Slave N

Bits	Description
31:0	**ProgMMIORange1Addr**. Read-write. Reset: 0. MMIO decode base address for Range 1.

eSPIx58 eSPI MMIOTarget Range Register 2 for Slave N

Bits	Description
31:0	**ProgMMIORange2Addr**. Read-write. Reset: 0. MMIO decode base address for Range 2.

eSPIx5C eSPI MMIOTarget Range Register 3 for Slave N

Bits	Description
31:0	**ProgMMIORange3Addr**. Read-write. Reset: 0. MMIO decode base address for Range 3.

eSPIx60 eSPI MMIOTarget Range Register 4 for Slave N

Bits	Description
31:16	**ProgMMIORange1Size**. Read-write. Reset: 0.
15:0	**ProgMMIORange0Size**. Read-write. Reset: 0.

eSPIx64 eSPI MMIOTarget Range Register 5 for Slave N

Bits	Description
31:16	**ProgMMIORange3Size**. Read-write. Reset: 0.
15:0	**ProgMMIORange2Size**. Read-write. Reset: 0.

eSPIx68 eSPI Slave N Configuration

Bits	Description
31	**CRCCheckingEnable.** Read-write. Reset: 0. This bit is set to 1 by eSPI master to enable the CRC checking on the eSPI bus. 0=CRC checking is disabled. 1=CRC checking is enabled.
30	**AlertMode**. Read-write. Reset: 0. This bit configures the Alert mechanism used by the slave to initiate a transaction on the eSPI interface. 0=I/O[1] pin is used to signal the Alert event. 1=A dedicated Alert# pin is used to signal the Alert event. This bit can only be '0' in a single master-single slave topology. For single master-multiple slave topology, this bit must be programmed to '1'.
29:28	**IOModeSelect**. Read-write. Reset: 0. eSPI master programs this field to enable the appropriate mode of operation, which takes effect at the deassertion edge of the chip select#. The IO mode configured in this field must be supported by both the master and the slave. Single IO mode is supported by default: Bits — Definition 00b — Single IO. 01b — Dual IO. 10b — Quad IO. 11b — Reserved.
27:25	**OperatingFreq**. Read-write. Reset: 0. All the slave devices will run in the same frequency of eSPI Bus clock, and only slave0 "Operating Frequency" field is valid, other slaves are reserved for this field. This field identifies the frequency of operation: Bits — Definition 000b — 16.6 MHz. 001b — 33 MHz. 010b — 66 MHz. 111b-011b — Reserved.
24:5	Reserved.
4	**OOBValidBitEnable**. Read-write. Reset: 0.
3	**PChannelEnable**. Read-write. Reset: 0. 1=Enable Peripheral Channel.
2	**VWChannelEnable**. Read-write. Reset: 0.
1	**OOBMessageChannelEnable**. Read-write. Reset: 0.
0	**FlashAccessChannelEnable**. Read-write. Reset: 0.

eSPIx6C eSPI Slave N Interrupt Enable

Bits	Description
31	Reserved.
30	**UpstreamOOBSuccessReceiveEn: OOB Message Received Enable**. Read-write. Reset: 0.
29	**UpstreamPSuccessReceiveEn: Peripheral Message Received Enable**. Read-write. Reset: 0.
28	**DownstreamProgSuccessCommandInterruptEn: Downstream Register Command Complete Enable**. Read-write. Reset: 0.
27	**IndexGrp3ReceivedEn: Virtual Wire Index Group 3 Received Enable**. Read-write. Reset: 0.
26	**IndexGrp2ReceivedEn: Virtual Wire Index Group 2 Received Enable**. Read-write. Reset: 0.
25	**IndexGrp1ReceivedEn: Virtual Wire Index Group 1 Received Enable**. Read-write. Reset: 0.
24	**IndexGrp0ReceivedEn: Virtual Wire Index Group 0 Received Enable**. Read-write. Reset: 0.
23:20	Reserved.
19	**AlinkTimeoutEn: Alink Bus Watch Dog Timer Timeout Enable**. Read-write. Reset: 0.
18	**ALinkMsterAbortEn: Alink Bus Master Abort Enable**. Read-write. Reset: 0.
17:16	Reserved.
15	**ProtocolErrdetectEn: Protocol ERROR detected Enable**. Read-write. Reset: 0.
14	Reserved.
13	**PPktDataOver32ByteErrorInterruptEn: Peripheral Message Data Length Over 32 bytes Enable**. Read-write. Reset: 0.
12	**OOBPktDataOver32ByteErrorInterruptEn: OOB Packet Data Length Over 32 bytes Enable**. Read-write. Reset: 0.
11	**IllegalLengthEn: Illegal Response Length Received Enable**. Read-write. Reset: 0.
10	**IllegalTagEn: Illegal Response Tag Received Enable**. Read-write. Reset: 0.
9	**UnsuccessfulCPLEn: Unsuccessful CPL Received Enable**. Read-write. Reset: 0.
8	**UnrecongCycleTypeEn: Unrecognized Cycle Type Received Enable**. Read-write. Reset: 0.
7	**UnrecongCodeEn: Unrecognized Response Code Received Enable**. Read-write. Reset: 0.
6	**NonFatalErrCodeEn**. Read-write. Reset: 0.
5	**FatalErrCodeEn**. Read-write. Reset: 0.
4	**NoresponseCodeEn: Receive NO_RESPONSE Code and Enable interrupt**. Read-write. Reset: 0.
3	Reserved.
2	**CRCErrDetectEn: CRC Error detected Enable**. Read-write. Reset: 0.
1	**WaitStateTimeoutEn: eSPI Bus Wait State Insertion Max Out Enable**. Read-write. Reset: 0.
0	**BusTimingErrEn: eSPI Bus Timing Error Enable**. Read-write. Reset: 0.

eSPIx70 eSPI Slave N Interrupt Status

Bits	Description
31	Reserved.

30	**UpstreamOOBMsgRec.** Read-only. Reset: 0.
29	**UpstreamPMsgRec**. Read-only. Reset: 0.
28	**DownstreamProgSuccess**. Read-only. Reset: 0.
27	**IndexGrp3Received: Virtual Wire Index Group 3 Received Status**. Read-only. Reset: 0.
26	**IndexGrp2Received: Virtual Wire Index Group 2 Received Status**. Read-only. Reset: 0.
25	**IndexGrp1Received: Virtual Wire Index Group 1 Received Status**. Read-only. Reset: 0.
24	**IndexGrp0Received: Virtual Wire Index Group 0 Received Status**. Read-only. Reset: 0.
23:20	Reserved.
19	**ALinkBusWatchDogTimerTimeout**. Read-only Reset: 0.
18	**ALinkBusMaterAbort**. Read-only. Reset: 0.
17:16	Reserved.
15	**ProtocolErrDet: Protocol ERROR detected Status**. Read-only Reset: 0.
14	Reserved.
13	**PPktDataOver32ByteStatus**. Read-only. Reset: 0.
12	**OOBPktDataOver32ByteStatus**. Read-only. Reset: 0.
11	**IllegalLengthDet: Illegal Response Length Received Status**. Read-only. Reset: 0.
10	**IllegalTagDet: Illegal Response Tag Received Status**. Read-only. Reset: 0.
9	**FailCPLReceivedDet: Unsuccessful CPL Received Status**. Read-only. Reset: 0.
8	**UnrecognizedCycTypeDet**. Read-only. Reset: 0.
7	**UnregconizedResponseDet**. Read-only. Reset: 0.
6	**NonFatalErrDet**. Read-only. Reset: 0.
5	**FatalErrDet**. Read-only. Reset: 0.
4	**NoResponseDet**. Read-only. Reset: 0.
3	Reserved.
2	**CRCErrDet**. Read-only. Reset: 0.
1	**WaitStateTimeoutStatus:eSPI Bus Wait State Inseseration Max Out Status**. Read-only. Reset: 0.
0	**BusTimingErrStatus: Status of eSPI Bus Timing Error bit**. Read-only. Reset: 0.

eSPIx74 eSPI Slave N Received Peripheral Message Register 0

Bits	Description
31:24	**PMsgByte3**. Read-only. Reset: 0. Received Peripheral Message code.
23:16	**PMsgByte2**. Read-only. Reset: 0. Received Peripheral Message Length[7:0].
15:8	**PMsgByte1**. Read-only. Reset: 0. Tag[3:0] and Length[3:0].
7:0	**PMsgByte0**. Read-only. Reset: 0. Peripheral Message data.

eSPIx78 eSPI Slave N Received Peripheral Message Register 1

Bits	Description
31:24	**PMsgSpecificByte3: Peripheral Message Specific Byte3**. Read-only. Reset: 0.
23:16	**PMsgSpecificByte2: Peripheral Message Specific Byte2**. Read-only. Reset: 0.
15:8	**PMsgSpecificByte1: Peripheral Message Specific Byte1**. Read-only. Reset: 0.
7:0	**PMsgSpecificByte0: Peripheral Message Specific Byte0**. Read-only. Reset: 0.

eSPIx[98:7C:Step8] eSPI Slave N Received Peripheral Message Register 2-9

Bits	Description
31:0	**PMsgData: Peripheral Message Data**. Read-only. Reset: 0.

eSPIx9C eSPI Slave N Received Virtual Wires Register

Bits	Description
31:20	Reserved.
19	**HostRstAckIdx6: Host Reset Acknowledge**. Read-only. Reset: 0. Sent by Slave to acknowledge received HOST_RST_WARN virtual wire.
18	**RstCPUInitIdx6: Reset CPU INIT**. Read-only. Reset: 1. Send to request CPU reset on behalf of the Keyboard controller.
17	**SMIIdx6: System Management Interrupt**. Read-only. Reset: 1. Sent as general Purpose alert resulting in SMI code being invoked by BIOS.
16	**SCIIdx6: System Control Interrupt**:. Read-only. Reset: 1. Sent as general Purpose alert resulting in ACPI method being invoked by OS.
15	**Slave_Boot_Ld_Status: Slave Boot Load Status**. Read-only. Reset: 0. Sent by EC or BMC upon completion of Slave Boot Load from the master attached flash. 0=The boot image is corrupted, incomplete or otherwise unusable. 1=The boot code load was successful and that the integrity of the image is intact, or the boot code load from master attached flash is not required. The Slave_Boot_Load_Status must be sent in either the same or a previous virtual wire message as the Slave_Boot_Load_Done.
14	**NonFatalErrdx5**. Read-only. Reset: 0. Sent when a non-fatal error is detected not due to eSPI transaction on the bus. Non-fatal Error due to transaction on eSPI bus is signaled through RSP phase.
13	**FatalErrdx5**. Read-only. Reset: 0. Sent when a fatal error is detected not due to eSPI transaction on the bus. Fatal error due to transaction on eSPI bus is signaled through RSP phase.
12	**SlaveBootLoadDoneIdx5**. Read-only. Reset: 0. Sent when EC or BMC has completed its boot process as indication to eSPI master to continue with the G3 to S0 exit.
11	**PMEIdx4**. Read-only. Reset: 1. PCI Power Management Event, Shared by multiple eSPI.
10	**WakeIdx4**. Read-only. Reset: 1. Wake the Host from Sx on any event(WAKE#).
9	Reserved.
8	**OOBRstAckIdx4**. Read-only. Reset: 0. Sent by Slave to acknowledge received OOB_RST_ACK virtual wire from Host.

7:5	**IRQStatus**. Read-only. Reset: 0. Bits — Definition 000b — IRQ keep 0 unchanged. 001b — IRQ keep 1 unchanged. 010b — Clear IRQ, change from 1 to 0. 011b — Set IRQ, change from 0 to 1. 100b — IRQ change from Low/High/Low. 101b — IRQ change from High/Low/High. 110b — Reserved. 111b — Reserved.
4:0	**IRQSelection**. Read-only. Reset: 0. Bits — Definition 0_0000b — IRQ0. 0_0001b — IRQ1. 0_0010b — IRQ2. 1_0110b-0_0011b — Reserved. 1_0111b — IRQ23. 1_1111b-1_1000b — Reserved.

eSPIxA0 eSPI Slave N Virtual Wire Received Data Register

Bits	Description
31:24	**Group3VWData: Group3 Virtual Wire Data Register**. Read-only. Reset: 0. See Group0VWData.
23:16	**Group2VWData: Group2 Virtual Wire Data Register**. Read-only. Reset: 0. See Group0VWData.
15:8	**Group1VWData: Group1 Virtual Wire Data Register**. Read-only. Reset: 0. See Group0VWData.
7:0	**Group0VWData: Group0 Virtual Wire Data Register**. Read-only. Reset: 0. When VW MISC CONTRL register bit[0] is set, the eSPI master checks each received VW Index. If the received Index matches the "Group0 Virtual Wire Index Selection Register" (eSPIxA4[Group0VWIdxSelectReg]), the eSPI Master then updates this field with the new received value.

eSPIxA4 eSPI Slave N Virtual Wire Index Selection Register

Bits	Description
31:24	**Group3VWIdxSelectReg: Group3 Virtual Wire Index Selection Register**. Read-write Reset: 0.
23:16	**Group2VWIdxSelectReg: Group2 Virtual Wire Index Selection Register**. Read-write. Reset: 0.
15:8	**Group1VWIdxSelectReg: Group1 Virtual Wire Index Selection Register**. Read-write. Reset: 0.
7:0	**Group0VWIdxSelectReg: Group0 Virtual Wire Index Selection Register**. Read-write. Reset: 0.

eSPIxA8 eSPI Slave N Virtual Wire MISC Control Register

Bits	Description
31	**IRQ23Mask**. Read-write. Reset: 0. See IRQ0Mask.
30	**IRQ22Mask**. Read-write. Reset: 0. See IRQ0Mask.
29	**IRQ21Mask**. Read-write. Reset: 0. See IRQ0Mask.

28	**IRQ20Mask**. Read-write. Reset: 0. See IRQ0Mask.
27	**IRQ19Mask**. Read-write. Reset: 0. See IRQ0Mask.
26	**IRQ18Mask**. Read-write. Reset: 0. See IRQ0Mask.
25	**IRQ17Mask**. Read-write. Reset: 0. See IRQ0Mask.
24	**IRQ16Mask**. Read-write. Reset: 0. See IRQ0Mask.
23	**IRQ15Mask**. Read-write. Reset: 0. See IRQ0Mask.
22	**IRQ14Mask**. Read-write. Reset: 0. See IRQ0Mask.
21	**IRQ13Mask**. Read-write. Reset: 0. See IRQ0Mask.
20	**IRQ12Mask**. Read-write. Reset: 0. See IRQ0Mask.
19	**IRQ11Mask**. Read-write. Reset: 0. See IRQ0Mask.
18	**IRQ10Mask**. Read-write. Reset: 0. See IRQ0Mask.
17	**IRQ9Mask**. Read-write. Reset: 0. See IRQ0Mask.
16	**IRQ8Mask**. Read-write. Reset: 0. See IRQ0Mask.
15	**IRQ7Mask**. Read-write. Reset: 0. See IRQ0Mask.
14	**IRQ6Mask**. Read-write. Reset: 0. See IRQ0Mask.
13	**IRQ5Mask**. Read-write. Reset: 0. See IRQ0Mask.
12	**IRQ4Mask**. Read-write. Reset: 0. See IRQ0Mask.
11	**IRQ3Mask**. Read-write. Reset: 0. See IRQ0Mask.
10	**IRQ2Mask**. Read-write. Reset: 0. See IRQ0Mask.
9	**IRQ1Mask**. Read-write. Reset: 0. See IRQ0Mask.
8	**IRQ0Mask**. Read-write. Reset: 0. 1=IRQn received from Virtual Wire packet is masked and eSPI_IRQn is not sent to ACPI. 0=eSPI_IRQn is sent to ACPI.
7:5	Reserved.
4	**VWSusStatEnable**. Read-write. Reset: 0. Enable Hardware sending Virtual Wire packet when SUS_STAT# change. 1=Enable. 0=Disable.
3	**Group3VWEnable**. Read-write. Reset: 0.
2	**Group2VWEnable**. Read-write. Reset: 0.
1	**Group1VWEnable**. Read-write. Reset: 0.
0	**Group0VWEnable**. Read-write. Reset: 0. When Set, the VW channel checks the received Index. If the Index is same as the Group0 index register setting; VW stores the data into Group0 data register.

eSPIxAC eSPI Slave N Virtual Wires Polarity Register

Bits	Description
31:24	Reserved.
23:0	**IRQn_Polarity**. Read-only. Reset: 0. Set IRQn Polarity for IRQ[23:0]. 1=Do not invert VW IRQ packet data, just route the VW IRQ data to eSPI_ACPI_IRQxx_B directly. 0=Invert VW IRQ data before routing to eSPI_ACPI_IRQxx_B. If slave IRQn is low active level interrupt or high active edge interrupt, no need to invert VW IRQ data, this bit must be set. If slave IRQ is high active level interrupt or low active edge interrupt, need to invert VW IRQ data, this bit must be cleared.

eSPIxB0 eSPI Slave N Received OOB Message Register 0

Bits	Description
31:24	**OOBPktByte3**. Read-only. Reset: 0. SMBus Slave Address. Bit[0]=1.
23:16	**OOBPktByte2**. Read-only. Reset: 0. Length[7:0].
15:8	**OOBPktByte1**. Read-only. Reset: 0. [15:12]: Tag. [11:8]: Length[11:8].
7:0	**OOBPktByte0**. Read-only. Reset: 0. Cycle Type for OOB Msg.

eSPIxB4 eSPI Slave N Received OOB Message Register 1

Bits	Description
31:24	**OOBPktByte6**. Read-only. Reset: 0. SMBus Data Byte 1 if OOB byte count is larger than 1.
23:16	**OOBPktByte6**. Read-only. Reset: 0. SMBus Data Byte 0.
15:8	**OOBPktByte5**. Read-only. Reset: 0.SMBus Byte Count. SMBUS spec states “A Block Read or Write is allowed to transfer a maximum of 32 data bytes”.
7:0	**OOBPktByte4**. Read-only. Reset: 0. SMBus OP Code.

eSPIx[D0:B8:Step7] eSPI Slave N Received OOB Message Register 2-8

Bits	Description
31:24	**SMBusDataByte5**. Read-only. Reset: 0.
23:16	**SMBusDataByte4**. Read-only. Reset: 0.
15:8	**SMBusDataByte3**. Read-only. Reset: 0.
7:0	**SMBusDataByte2**. Read-only. Reset: 0.

eSPIxD4 eSPI Slave N Received OOB Message Register 9

Bits	Description
31:25	Reserved.
24	**OOBMsgValid**. Read-only. Reset: 0.
23:16	**SMBusDataByte32**. Read-only. Reset: 0. If OOB byte count == 31, this field is used to store the optional PEC bytes.
15:8	**SMBusDataByte31**. Read-only. Reset: 0.
7:0	**SMBusDataByte30**. Read-only. Reset: 0.

eSPIxD8 eSPI Slave N Reserved Register 0

Bits	Description
31:0	Reserved.

eSPIxDC eSPI Slave N Reserved Register 1

Bits	Description
31:0	Reserved.

3.26.9 High Precision Event Timer (HPET) Registers

HPET registers are accessed through two methods:
- Memory access to HPET memory address range from FED0_0000h to FED0_01FFh. Program PMx00[HpetEn] = 1 to enable HPET decoding.
- Memory mapped access through the AcpiMmio region. The HPET registers range from FED8_0000h+C00h to FED8_0000h+CFFh. See PMx04[MmioEn].

HPETx000 ID

Bits	Description
31:16	**VendorID**. Read-only. Reset: 1022h.
15	**LegacyCap**. Read-only. Reset: 1. Legacy replacement interrupt is supported.
14	Reserved.
13	**CounterSizeCap**. Read-only. Reset: 0. 0=32 bits wide. 1=64 bits wide.
12:8	**NumTmrCap**. Read-only. Reset: 02h. Three timers are supported.
7:0	**RevID**. Read-only. Reset: 01h. Revision ID.

HPETx004 ClkPeriod

Bits	Description
31:0	**CounterClkPeriod**. Read-only; Updated-by-hardware. Reset: 0429_B17Eh. Specifies the clock period of each HPET timer tick. HPET main counter runs at 14.31818 MHz. The unit is femtosecond (10^-15 seconds). The value of this register can be modified through MISCx74.

HPETx010 Config

Bits	Description
31:2	Reserved.
1	**LegacyEn**. Read-write. Reset: 0. 1=Timer0 interrupt goes to IRQ0 of PIC controller, INT2 of IOAPIC; Timer1 interrupt goes to IRQ8 of PIC controller, INT8 of IOAPIC.
0	**TmrEn**. Read-write. Reset: 0. 0=Pause main counter and disable all timer interrupts. 1=Allow main counter to run and allow timer interrupts if enabled.

HPETx020 Interrupt Status

Bits	Description
31:3	Reserved.
2	**Tmr2IntrSts**. IF (HPETx140[TmrIntTyp] == 1) THEN Read; Set-by-hardware; Write-1-to-clear. ELSE Read-write. ENDIF. Reset: 0. 0=Timer2 interrupt is not active. 1=Timer2 interrupt is active. If Timer2 is set to edge-triggered mode, software should ignore this bit and always write 0 to this bit.

1	**Tmr1IntrSts**. IF (HPETx120[TmrIntTyp] == 1) THEN Read; Set-by-hardware; Write-1-to-clear. ELSE Read-write. ENDIF. Reset: 0. 0=Timer1 interrupt is not active. 1=Timer1 interrupt is active. If Timer1 is set to edge-triggered mode, software should ignore this bit and always write 0 to this bit.
0	**Tmr0IntrSts**. IF (HPETx100[TmrIntTyp] == 1) THEN Read; Set-by-hardware; Write-1-to-clear. ELSE Read-write. ENDIF. Reset: 0. 0=Timer0 interrupt is not active. 1=Timer0 interrupt is active. If Timer0 is set to edge-triggered mode, software should ignore this bit and always write 0 to this bit.

HPETx0F0 Main Counter

Bits	Description
63:32	**MainCounterHi**. Read-write. Reset: 0. IF (PMx00[HpetWidthSel] == 1) THEN Specifies the upper 32 bits of the 64-bit HPET main counter. Bits[31:0] contain the lower 32 bits of the 64-bit HPET main counter. Should be written to only when it is halted. ELSE Reserved. ENDIF.
31:0	**MainCounter**. Read-write. Reset: 0. Specified the HPET main counter, incremented by 1 on every clock. Should be written to only when it is halted.

HPETx1[4:0:Step2]0 Timer[2:0] Config Capability

Bits	Description
63:32	**TmrIntRouteCap**. Read-only. Reset: 00C0_0000h. Indicates which INT entry of IOAPIC can be assigned to the timer interrupt.
31:16	Reserved.
15	**TmrFsbCap**. Read-only. Reset: 1. 1=Front side bus delivery is supported.
14	**TmrFsbEn**. Read-write. Reset: 0. 1=Enable front side bus delivery of interrupt.
13:9	**TmrIntRoute**. Read-write. Reset: 0. This specifies which INT entry of IOAPIC the timer is routed to if HPETx010[LegacyEn] == 0.
8	**Tmr32ModeEn**. Read-only. Reset: 0. 0=64-bit timer is not supported. 1=64-bit timer is supported.
7	Reserved.
6	**TmrSetPer**. Read-write; Updated-by-hardware. Reset: 0. 1=Allow software to set the timer's accumulator if the timer is set to periodic mode. The bit is automatically cleared when HPETx1[4:0:Step2]8 is written by software.
5	**TmrSizeCap**. Read-only. Reset: 0. The timer is 32 bits wide.
4	**TmrTypCap**. Read-only. Reset: 1. The timer supports periodic interrupt delivery mode.
3	**TmrTyp**. Read-write. Reset: 0. Selects the timer interrupt type. 0=Non-periodic. 1=Periodic.
2	**TmrIntEn**. Read-write. Reset: 0. 1=Enable the timer interrupt.
1	**TmrIntTyp**. Read-write. Reset: 0. Specifies the timer interrupt polarity. 0=Edge triggered. 1=Level triggered.
0	Reserved.

HPETx1[4:0:Step2]8 Timer[2:0] Comparator

Bits	Description
63:32	**ComparatorHi**. Read-write. IF (PMx00[HpetWidthSel] == 1) THEN Reset: FFFF_FFFFh. ELSE Reset: 0000_0000h. ENDIF. Specifies the upper 32 bits of the 64-bit timer comparator. Bits[31:0] contain the lower 32 bits of the 64-bit timer comparator.
31:0	**Comparator**. IF (HPETx1[4:0:Step2]0[TmrTyp] == 0) THEN Read-write. ELSEIF ((HPETx1[4:0:Step2]0[TmrTyp] == 1) && (HPETx1[4:0:Step2]0[TmrSetPer] == 1)) THEN Read-write. ELSE Read-only. ENDIF. Reset: FFFF_FFFFh. Comparator is periodically incremented by the value last written to this register. This register is updated by hardware when (HPETx1[4:0:Step2]0[TmrTyp] == 1) && the current value in this register matches with the value in HPETx0F0.

HPETx1[5:1:step2]0 Timer[2:0] FSB Interrupt Data

Bits	Description
31:0	**TnFsbIntVal**. Read-write. Reset: 0000_0000h. Software sets this 32-bit field to specify the write data of Front Side Bus (FSB) Interrupt Message.

HPETx1[5:1:step2]4 Timer[2:0] FSB Interrupt Address

Bits	Description
31:0	**TnFsbIntAddr**. Read-write. Reset: 0000_0000h. Software sets this 32-bit field to specify the address of Front Side Bus (FSB) Interrupt Message.

HPETx1[D:B]0 Timer[2:0] Comparator Base Shadow

Bits	Description
63:32	**TmrCompBaseShadowHi**. Read-write. IF (PMx00[HpetWidthSel] == 1) THEN Reset: FFFF_FFFFh. ELSE Reset: 0000_0000h. ENDIF. This is shadow of the base value of HPETx1[4:0:Step2]8[ComparatorHi]. Reading this register returns the base value of HPETx1[4:0:Step2]8[ComparatorHi]. Writing the register changes the base value of HPETx1[4:0:Step2]8[ComparatorHi].
31:0	**TmrCompBaseShadowLo**. Read-write. Reset: FFFF_FFFFh. This is shadow of the base value of HPETx1[4:0:Step2]8[Comparator]. Reading this register returns the base value of HPETx1[4:0:Step2]8[Comparator]. Writing the register changes the base value of HPETx1[4:0:Step2]8[Comparator].

HPETx1[D:B]8 Timer[2:0] Comparator Shadow

Bits	Description
63:32	**TmrCompShadowHi**. Read-write. IF (PMx00[HpetWidthSel] == 1) THEN Reset: FFFF_FFFFh. ELSE Reset: 0000_0000h. ENDIF. This is shadow of the current value of HPETx1[4:0:Step2]8[ComparatorHi]. Reading this register returns the current value of HPETx1[4:0:Step2]8[ComparatorHi]. Writing the register changes the current value of HPETx1[4:0:Step2]8[ComparatorHi].
31:0	**TmrCompShadowLo**. Read-write. Reset: FFFF_FFFFh. This is shadow of the current value of HPETx1[4:0:Step2]8[Comparator]. Reading this register returns the current value of HPETx1[4:0:Step2]8[Comparator]. Writing the register changes the current value of HPETx1[4:0:Step2]8[Comparator].

HPETx1E0 Main Counter RTC

Bits	Description
63:32	**MainCounterRtcHi**. Read-write. Reset: 0000_0000. IF (PMx00[HpetWidthSel] == 1) THEN Specifies the shadow of HPETx0F0[MainCounterHi]. See MainCounterRtcLo. ELSE Reserved. ENDIF.
31:0	**MainCounterRtcLo**. Read-write. Reset: 0000_0000. This is a shadow of HPETx0F0[MainCounter]. It samples the value of Main Counter at every falling edge of the Real-Time Clock (RTC) 32 KHz clock for software to read. When software writes this register, HPETx0F0[MainCounter] is updated with the same value written to this register, and then enabled counting at the next RTC clock falling edge. The purpose of this register is for the convenience of software save/restore HPET.

HPETx1E8 Next Timer Remain

Bits	Description
63:32	**NxtTmrRemainHi**. Read-write. Reset: 0000_0000. IF (PMx00[HpetWidthSel] == 1) THEN Specifies the upper 32 bits of the timer ticks remaining before the next enabled comparator interrupt. ELSE Reserved. ENDIF.
31:0	**NxtTmrRemainLo**. Read-only. Reset: FFFF_FFFFh. Specifies how many timer ticks remaining before the next enabled comparator interrupt.

3.26.10 Miscellaneous (MISC) Registers

MISC register space is accessed through the AcpiMmio region. The MISC registers range from FED8_0000h+E00h to FED8_0000h+EFFh. See PMx04[Mmio].

MISCx00 GPPClkCntrl

BIOS: See 2.17.8.1.

Bits	Description
31	Reserved.
30:27	**CG1PLL_VREG_BIAS**. Read-write. Reset: 0. Voltage regulator current bias control (Spare).
26:25	**CG1PLL_VREG_CNTL**. Read-write. Reset: 0. Voltage regulator voltage control (Spare).
24	**CG1PLL_VREG_VREF_SEL**. Read-write. Reset: 0. Select Voltage regulator reference (Spare). 0=Bandgap. 1=Resistor Divider.
23	**CG1PLL_PLLCORE_PWDN_EN**. Read-write. Reset: 0. Control for powering down PLL's analog core (including PFD/CP, OTA, VCO, REFGENs, LF and all differential output drivers). It is only used for external clock-chip mode to bypass XTALIN 25 MHz to SATA differentially, when CG1_PLL is required to be off. 0=Disable. 1=Enable.
22:19	**CGPLL_BG_ADJ**. Read-write. Reset: 8h. Fine adjust for band-gap. Level shifted inside PLL to 1.8V.
18:16	Reserved.
15:12	**GPP_CLK3_ClockRequestMapping**. Read-write. Reset: Fh. GPP3 PCIe® clock pins (GPP_CLK3P/GPP_CLK3N) output control by CLK_REQ# pin. GPP_CLK3P/GPP_CLK3N pins are powered off when FCH is strapped to use an external clock, and powered on when FCH is strapped to operate in integrated clock mode. When FCH is in integrated clock mode, GPP3 PCIe clock can be powered off according to the CLK_REQ mapping table below, and the selected CLK_REQ# input can power off the GPP3 PCIe clock output pins if it is asserted. GPP3_CLKREQ_Mapping: Bits — Definition 0000b — Off. 0001b — CLK_REQ0#. 0010b — CLK_REQ1#. 0011b — CLK_REQ2#. 0100b — CLK_REQ3#. 1011b-0101b — Reserved; Off. 1100b — CLK_REQGfx0#. 1110b-1101b — Reserved; Off. 1111b — On.

11:8	**GPP_CLK2_ClockRequestMapping**. Read-write. Reset: Fh. GPP2 PCIe clock pins (GPP_CLK2P/GPP_CLK2N) output control by CLK_REQ# pin. GPP_CLK2P/GPP_CLK2N pins are powered off when FCH is strapped to use an external clock and powered on when FCH is strapped to operate in integrated clock mode. When FCH is in integrated clock mode, GPP2 PCIe clock can be powered off according to the CLK_REQ mapping table below and the selected CLK_REQ# input can power off the GPP2 PCIe clock output pins if it is asserted. GPP2_CLKREQ_Mapping: Bits — Definition 0000b — Off. 0001b — CLK_REQ0#. 0010b — CLK_REQ1#. 0011b — CLK_REQ2#. 0100b — CLK_REQ3#. 1011b-0101b — Reserved; Off. 1100b — CLK_REQGfx0#. 1110b-1101b — Reserved; Off. 1111b — On.

7:4	**GPP_CLK1_ClockRequestMapping**. Read-write. Reset: Fh. GPP1 PCIe clock pins (GPP_CLK1P/GPP_CLK1N) output control by CLK_REQ# pin. GPP_CLK1P/GPP_CLK1N pins are powered off when FCH is strapped to use an external clock and powered on when FCH is strapped to operate in integrated clock mode. When FCH is in integrated clock mode, GPP1 PCIe clock can be powered off according to the CLK_REQ mapping table below and the selected CLK_REQ# input can power off GPP1 PCIe clock output pins if it is asserted. GPP1_CLKREQ_Mapping: Bits Definition 0000b Off. 0001b CLK_REQ0#. 0010b CLK_REQ1#. 0011b CLK_REQ2#. 0100b CLK_REQ3#. 1011b-0101b Reserved; Off. 1100b CLK_REQGfx0#. 1110b-1101b Reserved; Off. 1111b On.
3:0	**GPP_CLK0_ClockRequestMapping**. Read-write. Reset: Fh. GPP0 PCIe clock pins (GPP_CLK0P/GPP_CLK0N) output control by CLK_REQ# pin. GPP_CLK0P/GPP_CLK0N pins are powered off when FCH is strapped to use an external clock and powered on when FCH is strapped to operate in integrated clock mode. When FCH is in integrated clock mode, GPP0 PCIe clock can be powered off according to the CLK_REQ mapping table below and the selected CLK_REQ# input can power off the GPP0 PCIe clock output pins if it is asserted. GPP0_CLKREQ_Mapping: Bits Definition 0000b Off. 0001b CLK_REQ0#. 0010b CLK_REQ1#. 0011b CLK_REQ2#. 0100b CLK_REQ3#. 1011b-0101b Reserved; Off. 1100b CLK_REQGfx0#. 1110b-1101b Reserved; Off. 1111b On.

MISCx04 ClkOutputCntrl

Bits	Description
31	**CG1_XTAL_DPLL_REFCLK_CLKEN**. Read-write. Reset: 0. 1=Enable CG1_PLL XTAL_DPLL_REFCLK Output.
30	**CG1_RegulatorOff**. Read-write. Reset: 0. 1=Turn off CG1_PLL Regulator.
29	**GFX0_CLK_output_override**. Read-write. Reset: 0. GFX0_CLK clock output override control. GFX0_CLK clock outputs are control by GFX0_CLK Clock Request mapping. Basically it depends on whether chip is in internal (LPCCLK1 == 1) or external (LPCCLK1 == 0) clock mode. This override bit allows to invert the strap value that controls clock output buffer. 0=Use GFX0_CLK Clock Request mapping to determine whether clock output buffer is on or off. 1=Invert the strap that controls those clock output buffer.

28	**PciePhyRefClkPwdn**. Read-write. Reset: 0. 1=Power-off Clkgen_outpad 100 Mhz CML reference clock for PCIE_PHY.
27	**CLKGEN_TestClkPwDn**. Read-write. Reset: 0. 1=Power off Clkgen_outpad TestClk Output buffer.
26	**EXT_CLK_ClockSourceOverride**. Read-write. Reset: 0. EXT_CLK clock source override control. EXT_CLK clock source is controlled by strap (LPCCLK1). If Strap (LPCCLK1) == 1, clock source is from CG_PLL. If Strap (LPCCLK1) == 0, clock source is from external clock chip through GPP_CLK3_P/N pins. This override bit allows to invert the strap that controls EXT_CLK clock source. 0=Use the strap value (LPCCLK1) to determine whether EXT_CLK clock source from either CG_PLL or external clock chip. 1=Invert the strap that controls EXT_CLK clock source.
25	Reserved.
24:23	**CG1PLL_DS_Order**. Read-write. Reset: 1. Delta-Sigma (DS) order setting. Bits Definition 00b disable 01b 1st order 10b 2nd order 11b 3rd order
22	**CG1PLL_DS_PrbsEn**. Read-write. Reset: 1. 1=Enable CG1_PLL PRBS17 generator in DS modulator.
21:20	**CG1PLL_FBDIV_FractionCtl**. Read-write. Reset: 1. CG1_PLL Slip Req/Ack handshaking control: Bits Definition 00b Slip_req/Slip_ack handshake (8 VCO clock cycles per slip) 01b Slip once every clock cycle (ignore slip_ack) 10b Slip once every 2 clock cycles (ignore slip_ack) 11b Slip once every 4 clock cycles (ignore slip_ack)
19:17	Reserved.
16	**ClockBufferBiasPowerDownEnable**. Read-write. Reset: 1. 1=Power down the clock buffers bias current circuit for power saving.
15:14	Reserved.
13	**PCIE_RCLK_PowerDownEnable**. Read-write. Reset: 0. BIOS: See 2.17.8.1. 1=Power down PCIE_RCLK input buffer for power saving.
12	**GFX_GPP_CLK_ClockOutputOverride: GFX_GPP_CLK [2:0] ClockOutputOverride**. Read-write. Reset: 0. GFX_GPP_CLK[2:0] clock output override control. GFX_CLK and GPP_CLK[2:0] clock outputs are controlled by [SLT_GFX_CLKClockRequestMapping] and MISCx00[GPP_CLK0_ClockRequestMapping, GPP_CLK1_ClockRequestMapping, GPP_CLK2_ClockRequestMapping]. Basically it depends on whether chip is in internal (Strap (LPCCLK1) == 1) or external (Strap (LPCCLK1) == 0) clock mode. This override bit allows to invert the strap value that controls clock output buffer. 0=Use SLT_GFX Clock Request mapping and GPP_CLK[2:0] Clock Request mapping to determine whether clock output buffer is on or off. 1=Invert the strap that controls those clock output buffer.
11	**GPP_CLK3_ClockOutputOverride**. Read-write. Reset: 0. GPP_CLK3 clock output override control. GPP_CLK3 is a bi-directional pin depending on LPCCLK1 strap value. If Strap (LPCCLK1) == 1, GPP_CLK3 provides clock for external device and output buffer will be on. If Strap (LPCCLK1) == 0, GPP_CLK3 receives clock from external clock chip and output buffer will be off. This override bit allows to invert the strap that controls GPP_CLK3 clock output buffer. 0=Use the strap value (LPCCLK1) to determine whether GPP_CLK3 clock output buffer is on or off. 1=Invert the strap that controls GPP_CLK3 clock output buffer.

10	**CPU_CLK_ClockSourceOverride**. Read-write. Reset: 0. CPU_CLK clock source override control. CPU_CLK clock source is controlled by strap (LPCCLK1). If Strap (LPCCLK1) == 1, CPU_CLK clock source is from CG_PLL. If Strap (LPCCLK1) == 0, CPU_CLK clock source is from external clock chip through GPP_CLK3_P/N pins. This override bit allows to invert the strap that controls CPU_CLK clock source. 0=Use the strap value (LPCCLK1) to determine whether CPU_CLK clock source from either CG_PLL or external clock chip. 1=Invert the strap that controls CPU_CLK clock source.
9:8	Reserved.
7:4	**SLT_GFX_CLK_ClockRequestMapping**. Read-write. Reset: Fh. Gfx PCIe® clock pins (SLT_GFX_CLKP/SLT_GFX_CLKN) output control. SLT_GFX_CLKP/SLT_GFX_CLKN pins are powered off when FCH is strapped to use an external clock, and powered on when FCH is strapped to operate in integrated clock mode. When FCH is in integrated clock mode, Gfx PCIe clock can be powered off according to the CLK_REQ mapping table below, and the selected CLK_REQ# input can power off Gfx PCIe clock output pins if it is asserted. Gfx_CLKREQ_Mapping: Bits — Definition 0000b — Off. 0001b — CLK_REQ0#. 0010b — CLK_REQ1#. 0011b — CLK_REQ2#. 0100b — CLK_REQ3#. 1011b-0101b — Reserved; Off. 1100b — CLK_REQGfx0#. 1110b-1101b — Reserved; Off. 1111b — On.
3:2	**CG1PLL_LOCK_FREQ_SEL**. Read-write. Reset: 0. CG1_PLL VCO centre frequency selection for calibration. Bits — Definition 00b — set VCO centre at 1200 MHz. 01b — set VCO centre at 1000 MHz. 10b — set VCO centre at 800 MHz. 11b — Reserved.
1	**CG1PLL_BYPASS_CAL**. Read-write. Reset: 0. CG1_PLL Calibration bypass enable. 0=Always do VCO calibration after a reset. 1=Bypass VCO calibration after a reset.
0	**CG1PLL_CAL_BYPASS_REFDIV_EN**. Read-write. Reset: 0. CG1_PLL Calibration reference divider bypass enable. 0=Reference divider set by MISCx10[CG1PLL_CAL_REFDIV] during calibration. 1=Bypass reference divider during calibration.

MISCx08 CGPLLConfig1

Bits	Description
31:16	**CG1PLL_DS_Frac**. Read-write. Reset: 0. CG1_PLL DS fractional setting.
15:6	Reserved.
5	**Cg1PllSataClkBypass**. Read-write. Reset: 0. SATA clock source frequency select from CG1_PLL. SATA reference clock output from CG1_PLL has an option for either output 100 Mhz or bypass CG1_PLL RefClk input. 0=PLL 100Mhz. 1=Bypass CG1_PLL RefClk input (48 Mhz or 100 Mhz).
4	Reserved.

3	**CG1_GFX0CLK_PULL_DOWN_override**. Read-write. Reset: 0. GFX0_CLK is a bi-directional pin and has pull down control. This bit is used to override pull-down control. 0=Pull down control logic determine if GFX0_CLK need to pull-down. 1=Invert the pull down control logic.
2	**CG1_GPP1CLK_PULL_DOWN_override**. Read-write. Reset: 0. GPP1_CLK is a bi-directional pin and has pull down control. This bit is used to override pull-down control. 0=Pull down control logic determine if GPP1_CLK need to pull-down. 1=Invert the pull down control logic.
1	**CG1_GPP0CLK_PULL_DOWN_override**. Read-write. Reset: 0. GPP0_CLK is a bi-directional pin and has pull down control. This bit is used to override pull-down control. 0=Pull down control logic determine if GPP0_CLK need to pull-down. 1=Invert the pull down control logic.
0	**CG1PLL_SpreadSpectrumEnable**. Read-write. Reset: 0. 1=Enable CG1_PLL Spread Spectrum. 0=Disable Spread Spectrum.

MISCx0C CGPLLConfig2

Bits	Description
31:25	**CG1PLL_PDIV_PCIeCLK**. Read-write. Reset: 0Ch. CG1_PLL 100 MHz clock post divider (PCIe® and SATA clock). Bits: Definition 00h: Bypass 01h: Bypass 7Fh-02h: Divide by <CG1PLL_PDIV_PCIeCLK>
24:18	**CG1PLL_PDIV_CoreCLK**. Read-write. Reset: 03h. CG1_PLL 400 MHz clock post divider (core clock). Bits: Definition 00h: Bypass 01h: Bypass 7Fh-02h: Divide by <CG1PLL_PDIV_CoreCLK>
17:16	**SATA_REFIN_ClkBufDrvStr**. Read-write. Reset: 01b. Specifies the drive strength control for internal SATA RefClk input differential clock buffers (from CLKGEN to SATA_PHY). Bits: Definition; Bits: Definition 00b: 6X driving; 10b: 10X driving 01b: 8X driving; 11b: 12X driving
15:14	**USB2_REFIN_ClkBufDrvStr**. Read-write. Reset: 01b. Specifies the drive strength control for internal USB2 RefClk input differential clock buffers (from CLKGEN to USB2_PHY). Bits: Definition; Bits: Definition 00b: 1.5X driving; 10b: 3X driving 01b: 2X driving; 11b: 3.5X driving
13:12	**USB3_REFIN_ClkBufDrvStr**. Read-write. Reset: 01b. Specifies the drive strength control for internal USB3 RefClk input differential clock buffers (from CLKGEN to USB3_PHY). Bits: Definition; Bits: Definition 00b: 2X driving; 10b: 4X driving 01b: 3X driving; 11b: 5X driving
11:10	**PCIe_REFIN_ClkBufDrvStr**. Read-write. Reset: 01b. Specifies the drive strength control for internal PCIe RefClk input differential clock buffers (from CLKGEN to PCIe). Bits: Definition; Bits: Definition 00b: 2X driving; 10b: 4X driving 01b: 3X driving; 11b: 5X driving

9:8	**CGPLL_REFIN_ClkBufDrvStr**. Read-write. Reset: 01b. Specifies the drive strength control for CGPLL external refclk input differential clock buffers (from CLKGEN to CGPLL). Bits Definition Bits Definition 00b 1.5X driving 10b 3X driving 01b 2X driving 11b 3.5X driving
7:6	**DPLL_REFIN_Clock_Driving_Strength**. Read-write. Reset: 01b. Drive Strength control for Internal DPLL RefClk Input differential Clock Buffers (from CLKGEN to DPLL). Bits Definition Bits Definition 00b 2X driving 10b 4X driving 01b 3X driving 11b 5X driving
5:2	Reserved.
1:0	**Cg1BgVrefBias**. Read-write. Reset: 0. Specifies the CGPLL voltage regulator current bias control from Bandgap VREG_BIAS[1:0]. Bits Definition Bits Definition 00b 100 uA 10b 75 uA 01b 120 uA 11b 85 uA

MISCx10 CGPLLConfig3

Bits	Description
31:30	Reserved.
29:26	**CG1PLL_CAL_REFDIV**. Read-write. Reset: 4h. Calibration reference divider control. It is only used when MISCx04[CG1PLL_CAL_BYPASS_REFDIV_EN] == 0. Bits Definition 0h Invalid 1h Invalid Fh-2h Divide by <CG1PLL_CAL_REFDIV>
25:22	**CG1PLL_FBDIV_Fraction**. Read-write. Reset: 0. Fractional part setting of the CG1_PLL feedback divider value. Bits Definition 0h 0 9h-1h <CG1PLL_FBDIV_Fraction*0.1> Fh-Ah 0
21:10	**CG1PLL_FBDIV**. Read-write. Reset: 032h. Integer part setting of the CG1_PLL Feedback divider value. Bits Definition 000h Divide by 2 001h Divide by 2 FFFh-002h Divide by <CG1PLL_FBDIV>
9:0	**CG1PLL_REFDIV**. Read-write. Reset: 002h. Reference divider control. It is only used after calibration is done. Bits Definition 000h Bypass 001h Bypass 3FFh-002h Divide by <CG1PLL_REFDIV>

MISCx14 CGPLLConfig4

Bits	Description
31:16	**CG1PLL_SS_AMOUNT_DSFRAC**. Read-write. Reset: 1FFFh. CG1_PLL Spread spectrum amount DS setting.
15:0	**CG1PLL_SS_STEP_SIZE_DSFRAC**. Read-write. Reset: 016Dh. CG1_PLL Spread spectrum step size DS control.

MISCx18 CGPLLConfig5

Bits	Description
31:27	Reserved.
26	**CG1PLL_VCO800M_EN**. Read-write. Cold reset: 0. 1=Enable CG1_PLL VCO running at 800 MHz. 0=Disable. A warm reset is required after setting this bit to enable/disable VCO running at 800 MHz.
25:24	**CG1PLL_PCALREF**. Read-write. Reset: 0. CG1_PLL VCO input2 control. Bits Definition 0h close loop configuration 1h 1/3 supply voltage 2h 1/2 supply voltage 3h 2/3 supply voltage
23:22	**CG1PLL_PVCOREF**. Read-write. Reset: 0. CG1_PLL VCO input1control. Bits Definition 0h close loop configuration 1h 1/3 supply voltage 2h 1/2 supply voltage 3h 2/3 supply voltage
21:13	Reserved.
12	**CG1PLL_SS_MODE**. Read-write. Reset: 1. CG1_PLL Spread spectrum mode select. 0=Center spread. 1=Down spread.
11:8	**CG1PLL_SS_AMOUNT_NFRAC_SLIP**. Read-write. Reset: 2. CG1_PLL Spread spectrum amount fractional setting.
7:0	**CG1PLL_SS_AMOUNT_FBDIV**. Read-write. Reset: 0. CG1_PLL Spread spectrum amount setting.

MISCx1C CGPLLConfig6

BIOS: See 2.17.8.3 [CG_PLL CMOS Clock Driver Setting for Power Saving].

Bits	Description
31:30	Reserved.
29	**CgpllClkDriverUpdate**. Read-write. Cold Reset: 1. CGPLL clock output drive type update. CGPLL has CML and CMOS type of clock output drivers. The selections are in MISCx1C[28:21]. 0=CG_PLL clock driver type selection only gets updated from MISCx1C[28:21] when CG_PLL got reset. 1=CG_PLL clock driver type selection gets updated on the fly from MISCx1C[28:21].

28:25	**Cg1RefClk48MHzDriverType**. Read-write. Cold Reset: 0010b. Selects CG1_PLL RefClk 48 Mhz clock driver type. For each bit, 0=CML type clock driver. 1=CMOS type clock driver (consume less power). Bit Definition [0] Reserved [1] Buf48 MHz for USB2 and SATA [2] Reserved [3] Reserved
24:21	**Cg1ClkDriverType**. Read-write. Cold Reset: 1011b. Selects CG1_PLL clock driver type. For each bit, 0=CML type clock driver. 1=CMOS type clock driver (consume less power). Bit Definition [0] USB [1] SATA [2] Reserved [3] PCIe®
20	**SataRefClkSrc**. Read-write. Reset: 0. BIOS: 1. Selects SATA reference clock source. 0=SATA reference clock is from CG XTAL 48 MHz non-spread. 1=Internal 100 MHz spread clock.
19:18	Reserved.
17:9	**CG1PLL_LF_MODE**. Read-write. Reset: 0FEh. CG1_PLL Loop filter control. It defines the setting of loop filter R and C values.
8	Reserved.
7	**Force25mXtalPadPwdn**. Read-write. Reset: 0. Force 25M XTAL Pad Power Down. Set this bit and MISCx40[Pwdn25Mxtal] to 1 to power down 25 MHz XTAL Pad.
6:5	Reserved.
4:0	**CG1PLL_CAL_MODE**. Read-write. Reset: 5. CG1_PLL VCO calibration mode control. Bits Definition 0_XXXXb VCO mode set to calibrated values. 1_XXXXb VCO mode forced to the value set in CG1PLL_CAL_MODE[3:0].

MISCx20 IMP Calibration

Bits	Description
31:26	**CG1PLL_SPARE**. Read-write. Reset: 0. CG1_PLL general spare pins.
25:23	Reserved.
22:20	**CG1PLL_DIG_SPARE**. Read-write. Reset: 0. CG1_PLL digital block spare pins.
19	**CG1PLL_SATA_CLK_CUSTOM_PDIV_EN**. Read-write. Reset: 1. CG1_PLL CLKOUT3B (SATA 100 MHz) Post-dividers selection. 0=Use Programmable CMOS PostDIV inside digital block. 1=Use Custom CMOS Post-dividers.
18	**CG1PLL_PCIE_CLK_CUSTOM_PDIV_EN**. Read-write. Reset: 1. CG1_PLL CLKOUT3A (PCIe® 100 MHz) post divider selection. 0=Use Programmable CMOS PostDIV inside digital block. 1=Use Custom CMOS Post-dividers.
17	**CG1PLL_USB2CLK_CUSTOM_PDIV_EN**. Read-write. Reset: 0. CG1_PLL CLKOUT2 (USB 48 MHz) post divider selection. 0=Use Programmable CMOS PostDIV inside digital block. 1=Use Custom CMOS Post-dividers.

16	**CG1PLL_CLKOUT3A_PDIVSEL**. Read-write. Reset:1. CG1_PLL PCIe clock select between RefClk (input of reference divider) and post divider 3. 0=Input of reference divider. 1=PDIV3 Output.
15	**CG1PLL_CLKOUT2_PDIVSEL**. Read-write. Reset: 1. CG1_PLL USB2 clock select between RefClk (input of reference divider) and post divider 2. 0=Input of reference divider. 1=PDIV2 Output.
14	**CG1PLL_CLKOUT1_PDIVSEL**. Read-write. Reset: 1. CG1_PLL Core Clock select between RefClk (input of reference divider) and post divider 1. 0=Input of reference divider. 1=PDIV1 Output.
13:12	**CG1PLL_PFD_PULSE_SEL**. Read-write. Reset: 2. CG1_PLL Change reset pulse inside PFD. Bits Definition 00b ~110 ps 01b ~220 ps 10b ~350 ps 11b ~500 ps
11:8	**CG1PLL_CP**. Read-write. Reset: 0001b. CG1_PLL charge pump control. Each step is 5 uA of additional current from 0 uA to 75 uA when CG1_PLL change from 0000b to 1111b.
7:0	**CG1PLL_IBIAS**. Read-write. Reset: 59h. CG1_PLL current bias control. Bit Definition [0] BIAS0 [1] BIAS1 [2] CPTUNE0 [3] CPTUNE1 [4] AMPTUNE0 (SPARE) [5] AMPTUNE1 (SPARE) [6] HALFGM [7] SPARE

MISCx24 ClkDrvStr1

Bits	Description
31:30	**DP_Link_Clock_Driving_Strength**. Read-write. Reset: 10b. Drive strength control for DP_Link differential Clock Buffers. Bits Definition 00b 12 mA 01b 13 mA 10b 14 mA 11b 15 mA
29:20	Reserved.
19:18	**SLT_GFX_ClockBufferDrivingStrengthControl**. Read-write. Reset: 10b. Drive strength control for SLT_GFX differential clock buffers. Bits Definition 00b 12 mA 01b 13 mA 10b 14 mA 11b 15 mA
17:8	Reserved.

7:6	**GppClk3_ClockBufferDrivingStrengthControl**. Read-write. Reset: 10b. Drive strength control for GPP_CLK3 differential clock buffers. Bits Definition 00b 12 mA 01b 13 mA 10b 14 mA 11b 15 mA
5:4	**GppClk2_ClockBufferDrivingStrengthControl**. Read-write. Reset: 10b. Drive strength control for GPP_CLK2 differential clock buffers. Bits Definition 00b 12 mA 01b 13 mA 10b 14 mA 11b 15 mA
3:2	**GppClk1_ClockBufferDrivingStrengthControl**. Read-write. Reset: 10b. Drive strength control for GPP_CLK1 differential clock buffers. Bits Definition 00b 12 mA 01b 13 mA 10b 14 mA 11b 15 mA
1:0	**GppClk0_ClockBufferDrivingStrengthControl**. Read-write. Reset: 10b. Drive strength control for GPP_CLK0 differential clock buffers. Bits Definition 00b 12 mA 01b 13 mA 10b 14 mA 11b 15 mA

MISCx28 ClkDrvStr2

Bits	Description
31	**USB3_RefClk_Pwdn**. Read-write. Reset: 0. Power Down USB3 Refclk (CML) Driver when in D3 while USB3 is Idle. 0=Clock on. 1=Power off.
30	**USB2_RefClk_Pwdn**. Read-write. Reset: 0. Power Down USB2 Refclk (CML) Driver when in D3 while USB2 is Idle. 0=Clock on. 1=Power off.
29	**SATA_RefClk_Pwdn**. Read-write. Reset: 0. Power Down SATA Refclk (CML) Driver when in D3 while SATA is Idle. 0=Clock on. 1=Power off.
28:22	Reserved.
21:19	**OSCOUT2_CLK_SEL**. Read-write. Reset: 0. Defines auxiliary output clock frequency on USB-CLK/OSCOUT2 pin. Bits Definition 000b 48 MHz, non-spread 001b 25 MHz, spread 111b-010b Reserved

18:16	**OSCOUT1_CLK_SEL**. Read-write. Reset: 0. Defines auxiliary output clock frequency on OSCOUT1 pin. Bits — Definition 000b — 48 MHz, non-spread 001b — 25 MHz, spread 111b-010b — Reserved
15:0	Reserved.

MISCx2C ClkGatedCntl

Bits	Description
31:28	Reserved.
27	**CG1PLL_UNLOCK_DET_EN**. Read-write. Cold reset: 0. Unlock detector enable (inside PLL MACRO).
26	**CG1PLL_LOCK_DETECTOR_SE**. Read-write. Cold reset: 1. Selects the mode of the PLL_LOCKED output pin. 0=New lock detector, 1=Legacy counter.
25:22	**CG1PLL_ANALOGOUT_MUX_CNTL**. Read-write. Cold reset: 0. ANALOGOUT_MUX_CNTL for mappllraw, selecting which signal is observable through BP_ANALOGOUT.
21:18	**CG1PLL_ANALOGIN_MUX_CNTL**. Read-write. Cold reset: 0. ANALOGIN_MUX_CNTL for mappllraw, selecting which signal is observable through BP_ANALOGIN.
17	**BlinkClkGateOffEn: B-Link Clock Gate Off Enable**. Read-write. Cold reset: 0. BIOS: See 2.17.8.2. Internal B-Link clock has two clock trees: one is a free running clock and the other is a gated clock. When all controllers agree to stop the gated B-Link clock and this bit is set, clock gating logic will gate off the clock tree from clock root. 0=Disable BLink Clock Gate Off function. 1=Enable BLink Clock Gate Off function.
16	**AlinkClkGateOffEn: A-Link Clock Gate Off Enable**. Read-write. Cold reset: 0. BIOS: See 2.17.8.2. Internal A-Link clock has two clock trees: one is a free-running clock and the other is a gated clock. When all controllers agree to stop the gated A-Link clock and this bit is set, clock gating logic will gate off the clock tree from clock root. 0=Disable A-Link Clock Gate-Off function. 1=Enable A-Link Clock Gate-Off function.
15:8	**BlinkClkGateOffThreshold**. Read-write. Reset: 0. B-Link Clock Gate Off Threshold. When all controllers agree to stop B-Link clock, clock gating logic will start a timer and deassert BLClk_Enable when the timer reaches a programmable threshold (BLClk_Enable is an internal handshake signal indicating whether gated B-Link clock is running or not). Bits — Definition 00h — Gated clock will stop at the 3rd falling edge after BLClk_Enable is deasserted. FFh-01h — Gated clock will stop at the <3+AlinkClkGateOffThreshold> falling edge after BLClk_Enable is deasserted.
7:0	**AlinkClkGateOffThreshold**. Read-write. Reset: 0. A-Link Clock Gate Off Threshold. When all controllers agree to stop A-Link clock, clock gating logic will start a timer and deassert ALClk_Enable when the timer reaches a programmable threshold (ALClk_Enable is an internal handshake signal indicating whether gated A-Link clock is running or not). Bits — Definition 00h — Gated clock will stop at the 3rd falling edge after ALClk_Enable is deasserted. FFh-01h — Gated clock will stop at the <3+AlinkClkGateOffThreshold> falling edge after ALClk_Enable is deasserted.

MISCx30 CGPLLConfig7

Bits	Description
31:0	Reserved.

MISCx34 CGPLLConfig8

Bits	Description
31:28	Reserved.
27	**DP_Link_Clock_PWDN**. Read-write. Reset: 0. DP_Link Refclk Clock Buffer Power Down control. 1=Power off.
26	**PCIE_Clock_PWDN**. Read-write. Reset: 0. FCH PCIe Refclk Clock Buffer Power Down control. 1=Power off.
25:0	Reserved.

MISCx38 CGPLLConfig9

Bits	Description
31:0	Reserved.

MISCx3C CGPLLConfig10

Bits	Description
31:0	Reserved.

MISCx40 MiscClkCntl1

Bits	Description
31	Reserved.
30	**CG1_Atomic_Update**. Read-write; Cleared-by-hardware. Reset: 0. 1=Request CG1_PLL to load FBDIV, FBDIV_Fraction, DS_FRAS value into CG1_PLL. The bit is cleared to 0 by hardware after request send to CG1_PLL.
29	**Atomic_Update_R_Skip**. Read-write. Reset: 0. 1=Hardware initializes CG1 and CG2 atomic update regardless whether CG_PLL is ready from previous update.
28:27	**SDCLK_PDIVSEL**. Read-write. Reset: 0. SDCLK Post divider selection.
26	**CG1PLL_FBDIV_Test**. Read-write. Cold reset: 0. 1=Enable loading MISCx10[CG1PLL_FBDIV] value for testing. A warm reset is required for this bit to take effect.
25:23	Reserved.
22	**AlinkClkSlow_Mode**. Read-write. Reset: 0. Enable internal A-Link clock running at slower speed for power saving. 1=Enable slow mode.

21	**AlinkClkSlowFreq**. Read-write. Reset: 0. Frequency to set A-Link clock speed. 0=50 Mhz. 1=33 Mhz.
20	**EnableBlinkClkSlowMode**. Read-write. Reset: 0. 1=Enable internal core clock (B-Link clock) running at slower speed (66 MHz) for power saving. This bit only takes effect when MISCx40[1] == 1.
19	**DrvStrAuxiliaryClk2**. Read-write. Reset: 1. Drive Strength Control for Auxiliary Clock2 (USBCLK/14M_25M_48M_50M_OSC). 0=4 mA. 1=8 mA.
18	**DrvStrAuxiliaryClk1**. Read-write. Reset: 1. Drive Strength Control for Auxiliary Clock1 (14M_25M_48M_50M_OSC). 0=4 mA. 1=8 mA.
17	**Pwdn25Mxtal**. Read-write. Reset: 0. 1=Power off 25M Xtal. 0=Power on 25M Xtal. Setting this bit to 1 can power off 25 MHz Xtal pad when the following conditions are not true: • Chip in integrated clock mode. • USB 48 MHz clock from CG2_PLL. • SATA use CG2_PLL clock as ref clock source. • 14.318 MHz clock generated from CG2_PLL. • 25 MHz Auxiliary clock selected.
16	Reserved.
15	**BlinkClkSlowEnable**. Read-write. Reset: 0. 1=Internal B-Link clock runs on 66 MHz. The normal internal B-Link clock runs on 133 MHz. See more detail on MISCx40[20].
14	**OscClkSwitchEn**. Read-write. Reset: 1. BIOS: 1. 1=The FCH uses the internal PLL to generate the 14 MHz OscClk.
13	**InvertTermResistor**. Read-write. Reset: 0. 1=Invert normal RefClk (NB_PCIE_CLK) termination. 0=Normal RefClk termination.
12	**Auxiliary_14mClk_Sel**. Read-write. Reset: 0. 14 MHz clock selection for Auxiliary clock output. 0=14.285 MHz clock is from CG2_PLL. 1=14.318 MHz clock is from CLK_REQG#/14M_OSC/GPIO65 pin.
11:8	Reserved.
7	**OSCOUT2_OutOff**. Read-write. Reset: 1. If USB uses internal 48 MHz as clock source, setting this bit to 1 will turn off USBCLK/OSCOUT2 clock output. 0=OSCOUT2 pin output enable. 1=OSCOUT2 pin output disable.
6	**UsbClkCfg**. Read-write. Reset: 1. Defines whether USB uses the internal or external 48 MHz clock. 0=External 48 MHz. 1=Internal 48 MHz. • If USB uses internal 48 MHz as clock source, USBCLK/OSCOUT2 pin can output 24 or 48 MHz clock depending on OSCOUT2_sel setting. • If USB uses external 48 MHz clock as clock source, USBCLK/OSCOUT2 pin cannot be used as auxiliary clock output. Output enable of OSCOUT2 pin is controlled by [UsbClkCfg] and [OSCOUT2_OutOff]. When [UsbClkCfg] == 1 & [OSCOUT2_OutOff] == 0, pin output enable will be on.
5	Reserved.
4	**Usb3RefclkSel**. Read-write. Reset: 0. BIOS: 0. USB3.0 reference clock source selection in internal clock mode. 0=100 MHz spread clock from internal CG1_PLL. 1=100 MHz non-spread clock from internal CG2_PLL.
3	Reserved.
2	**OSCOUT1_ClkOutputEnb**. Read-write. Reset: 1. 0=Enable Auxiliary Clock1 and OSCOUT1 clock output. 1=Disable.

1	**CoreSpeedMode**. Read-write. Reset: 0. Slow down Internal Core Clock (B-Link clock) for power saving. 0=Full speed B-Link clock. 1=Slow speed B-Link clock.
0	Reserved.

MISCx44 MiscClkCntl2

Bits	Description
31:10	Reserved.
9	**CG1PLL_USB2_CLKOUT_EN**. Read-write. Reset: 0. CG1_PLL USB2 28 MHz Clock Output Enable.
8	Reserved.
7	**CG1_PLL_REFCLK_RESET**. Read-write. Reset: 0.
6	**CG2PLL_ANTIGLITCH_RESET**. Read-write. Reset: 0. Anti-glitch reset: Forced to 1 during CG2_PLL reset and set to 0 after reset.
5	**CG1PLL_ANTIGLITCH_RESET**. Read-write. Reset: 0. Anti-glitch reset: Forced to 1 during CG1_PLL reset and set to 0 after reset.
4	Reserved.
3	**CG1_CAL_REFDIV_Update**. Read-write. Cold reset: 0. 1=Update CG1_CAL_REFDIV from Register MISCx10[CG1PLL_CAL_REFDIV]. A warm reset is required for this bit to take effect.
2:1	Reserved.
0	**CG1_FBDIV_LoadEn**. Read-write. Reset: 0. 1=Enable loading CG1PLL_FBDIV value from register MISCx10[CG1PLL_FBDIV].

MISCx48 MiscClkCntl3

Bits	Description
31:30	Reserved.
29:28	**CGPLL_XTAL_DPLL_DIFF_GAIN**. Read-write. Reset: 3. Diff CML Driver Strength Control for XTAL_DPLL_REFCLK_P/N (48 Mhz).
27:24	Reserved.
23:22	**CG1PLL_CLKOUT3B_DIFF_GAIN**. Read-write. Reset: 2. Diff CML Driver Strength Control for CG1PLL_CLKOUT3B_P/N.
21:20	**Cg1PllSataUsb2DiffGain**. Read-write. Reset: 2. Diff CML Driver Strength control for CG1PLL_SATA_USB2_P/N (48 Mhz).
19:18	**CG1PLL_CLKOUT3A_DPLL_DIFF_GAIN**. Read-write. Reset: 3. Diff CML Driver Strength Control for CG1PLL_DPLL_REFCLK_P/N.
17:16	**CG1PLL_CLKOUT3A_DIFF_PRE_GAIN**. Read-write. Reset: 3. Diff CML Pre-Driver Strength Control for CG1PLL_DPLL_REFCLK_P/N and CG1PLL_CLKGEN_PCIE_CPU_USB3_P/N.
15:14	**CG1PLL_CLKOUT3A_DIFF_GAIN**. Read-write. Reset: 2. Diff CML Driver Strength Control for CG1PLL_CLKGEN_PCIE_CPU_USB3_P/N
13:12	**CG1PLL_CLKOUT2_DIFF_GAIN**. Read-write. Reset: 2. Diff CML Driver Strength Control for CG1PLL_CLKGEN_USB2_P/N.

11:5	Reserved.
4:3	**Cg1XtalCmlPreDiffGain**. Read-write. Reset: 2. CG1_PLL CML Driver Strength selection for both SATA2/USB and DPLL_REFCLK output drivers.
2:0	Reserved.

MISCx4C MiscClkCntl4

Bits	Description
31	**LowPowerDisplay400MClkEnB**. Read-write. Reset: 0. 0=Enable low power display 400 Mhz clock output. 1=Disable.
30	**LowPowerDisplay300MClkEnB**. Read-write. Reset: 0. 0=Enable low power display 300 Mhz clock output. 1=Disable.
29:0	Reserved.

MISCx60 IdleCntrl

Bits	Description
31:24	**IdleCount**. Read-write. Reset: X. This returns the idle count from the latest monitored period.
23:16	Reserved.
15:8	**IdleThreshold**. Read-write. Reset: 80h. This defines the idle-ness threshold when the dynamic clock logic down shifts the clock frequency. Each unit represents 1/256 of the interval.
7:0	Reserved.

MISCx68 Memory Power Saving Control

Bits	Description
31:3	Reserved.
2	**ABBypassMemDsd**. Read-write. Reset: 1. BIOS: See 2.17.10.2. AB memory BypassMemdsd control. 0=Enable memory Deep Sleep and shutdown features. 1=Disable memory Deep Sleep and shutdown.
1:0	Reserved.

MISCx6C MiscControl

Bits	Description
31	**PostCodeWidthSel**. Read-write. Reset: 0. 1=Reading MISCx78 [PostCode] returns a 8-bit Post Code. 0=Reading MISCx78 [PostCode] returns a 32-bit Post Code.
30:0	Reserved.

MISCx70 OscFreqCounter

Bits	Description
31	**CountEnable**. Read-write. Reset: 0. 1=Enable the internal counter to count the number of OSC clocks. When software is not using this function, it should always set it back to 0 to conserve power.
30	**CountIsValid**. Read-only. Reset: 0. 1=OscCountPerSec is valid. Software should always wait for this bit to be set before it reads OscCountPerSec.
29:28	Reserved.
27:0	**OscCountPerSec**. Read-write. Reset: 000_0000h. Number of OSC clocks per 1 second. Whenever [CountEnable] is set, an internal counter starts counting the number of OSC clocks per second and record the count value here.

MISCx74 HpetClkPeriod

Bits	Description
31:0	**HpetClkPeriod**. Read-write. Reset: 0429_B17Eh. The register controls the value of HPETx004[CounterClkPeriod] register in HPET MMIO register space.

MISCx78 PostCode

Bits	Description
31:0	**PostCode[31:0]**. Read-only. Reset: 0. BIOS 32-bit writes to IO080 [PostCode 0] go to this internal 32-bit PostCode Register. Reads from IO080 [PostCode 0] return PostCode[7:0].

MISCx80 StrapStatus

Bits	Description
31:19	Reserved.
18	**BifGen2ComplianceStrap**. Read-only. Reset: 0. BIF gen 2 compliance strap.
17	**ClkGenStrap**. Read-only. Reset: X. BIOS: See 2.17.8. 1=Internal clocking mode; Use 48 MHz crystal clock as the reference clock. 0=External clocking mode; Use 100 MHz differential spread clock as the reference clock.
16	**BootFailTmrEnStrap**. Read-only. Reset: X. Enable Watchdog function.
15	**PciRomBootStrap**. Read-only. Reset: 1. PCI ROM Boot.
14:13	Reserved.
12	**CPUClkSelStrap**. Read-only. Reset: 0.
11	**ShortResetStrap**. Read-only. Reset: 0. Generate short reset.
10	**PciPllBypStrap**. Read-only. Reset: 0. Bypass PCI PLL (used in functional test at tester).
9	**SDPllBypStrap**. Read-only. Reset: 0.
8	Reserved.

7	**I2CRomStrap**. Read-only. Reset: 0. Getting UMI core strap from I^2C ROM or using default value.
6:3	Reserved.
2	**EcEnableStrap**. Read-only. Reset: 0. BIOS: See 2.17.7. 1=Integrated Micro-Controller (IMC) is enabled.
1	**UseLpcRomStrap**. Read-only. Reset: X.
0	Reserved.

MISCx90 AutoTransaction/Allow EC

Bits	Description
31:10	Reserved.
9	**DisableAuto**. Read; Write-1-only. Reset: 0. 1=The entire Auto Transaction logic is disabled. Once this bit is set, it cannot be cleared except by system reset.
8	**AllowECToAutoTransactEn**. Read-write. Reset: 0. Only BIOS can change this bit. 0=IMC cannot write to any of registers MISCx90/MISCx94/MISCx98/MISCx9C in the Auto Transaction Generation logic. 1=IMC can change any of these registers.
7:4	**TransactionType**. Read-write. Reset: 0. PCI Command type used for this auto transaction. See PCI specification for PCI command types.
3:2	**ByteCount**. Read-write. Reset: 0. Bits Definition 00b 1 byte 01b 2 bytes 10b 4 bytes 11b 4 bytes
1	**DualAddr**. Read-write. Reset: 0. 0=Use single address cycle. 1=Use dual address cycle.
0	**AutoExecute**. Read-write; Cleared-by-hardware. Reset: 0. Writing 1 to this bit causes the HW to execute the transaction defined by bits[7:1]. Once it is written, this bit stays as 1 until the transaction is completed, in which case it returns to 0.

MISCx94 AutoAddrLow

Bits	Description
31:0	**AutoAddrLow**. Read-write. Reset: 0. Low address to be used by the MISCx90[AutoExecute] operation.

MISCx98 AutoAddrHigh

Bits	Description
31:0	**AutoAddrHigh**. Read-write. Reset: 0. High address to be used by the MISCx90[AutoExecute] operation. This register is only applicable when MISCx90[DualAddr] == 1.

MISCx9C AutoData

Bits	Description
31:0	**AutoData**. Read-write. Reset: 0. If MISCx90[TransactionType] is read, this register returns the read data. If the MISCx90[TransactionType] is a write command, this register contains the write data. Note byte is aligned accordingly.

MISCxC0 CPU Pstate0

Bits	Description
31:15	Reserved.
14:12	**Core3Pstate**. Read-only. Reset: 0. See: Core0Pstate.
11	Reserved.
10:8	**Core2Pstate**. Read-only. Reset: 0. See: Core0Pstate.
7	Reserved.
6:4	**Core1Pstate**. Read-only. Reset: 0. See: Core0Pstate.
3	Reserved.
2:0	**Core0Pstate**. Read-only. Reset: 0. FCH will monitor the P-state of each CPU core. Bits Definition 000b P0 001b P1 010b P2 011b P3 100b P4 101b P5 110b P6 111b P7

MISCxC4 CPU Pstate1

Bits	Description
31:0	Reserved.

MISCxD0 CPU Cstate0

Bits	Description
31:15	Reserved.
14:12	**Core3Cstate**. Read-only. Reset: 0. See: Core0Cstate.
11	Reserved.
10:8	**Core2Cstate**. Read-only. Reset: 0. See: Core0Cstate.
7	Reserved.
6:4	**Core1Cstate**. Read-only. Reset: 0. See: Core0Cstate.

3	Reserved.
2:0	**Core0Cstate**. Read-only. Reset: 0. Bits Definition 000b C0 001b C1 010b C2 011b C3 100b C4 101b C5 110b C6 111b C7

MISCxD4 CPU Cstate1

Bits	Description
31:0	Reserved.

MISCxF0 SataPortSts

Bits	Description
31:26	Reserved.
25:24	**SataPortSel**. Read-write. Reset: 0. Bits Definition 00b Select “led” for Port 0 to 1 01b Select “det” for Port 0 to 1 10b Select “err” for Port 0 to 1 11b Select “led” for Port 0 to 1
23:2	Reserved.
1	**Port1Sts**. Read-write. Reset: X. The selected status of Port 1. This status bit indicates the internal status of SATA port 1.
0	**Port0Sts**. Read-write. Reset: X. The selected status of Port 0. This status bit indicates the internal status of SATA port 0.

MISCxF4 ClkCntrlSts

Bits	Description
31:0	Reserved.

3.26.11 GPIO Pin control registers

The GPIO pins are controlled by a combination of device enables and by their specific GPIO and IOMUX register pair.

3.26.11.1 GPIO Registers

GPIO registers are accessed through memory-mapped ACPIMMIO region. The offset is relative to FED8_0000h+1500h. GPIO bank 0 registers range from FED8_0000h+1500h to FED8_0000h+15FFh. GPIO Bank 1 registers range from FED8_0000h+1600h to FED8_0000h+16FFh. GPIO Bank 2 registers range from FED8_0000h+1700h to FED8_0000h+17FFh.

GPIOx[0F8:000:step4] GPIO Bank 0 Control Register

Each GPIO pin is controlled by 4 bytes. These registers control GPIO bank 0 pins: GPIO62-GPIO00.

Table 278: Reset Mapping for GPIOx[0F8:000:step4]

Register	Reset	Function	Register	Reset	Function
GPIOx000	0014_0000h	PWR_BTN_L_AGPIO0	GPIOx080	0000_0000h	Reserved
GPIOx004	0014_0000h	SYS_RESET_L_AGPIO1	GPIOx084	0000_0000h	Reserved
GPIOx008	0014_0000h	WAKE_L_AGPIO2	GPIOx088	0000_0000h	Reserved
GPIOx00C	0014_0000h	AGPIO3	GPIOx08C	0000_0000h	Reserved
GPIOx010	0024_0000h	AGPIO4	GPIOx090	0000_0000h	Reserved
GPIOx014	0004_0000h	AGPIO5_DEVSLP0_S5	GPIOx094	0000_0000h	Reserved
GPIOx018	0024_0000h	AGPIO6_LDT_RST_L	GPIOx098	0000_0000h	Reserved
GPIOx01C	0024_0000h	AGPIO7_LDT_PWROK	GPIOx09C	0024_0000h	VDDGFX_PD_AGPIO39
GPIOx020	0004_0000h	AGPIO8_SerPortTX_OUT	GPIOx0A0	0004_0000h	AGPIO40
GPIOx024	0024_0000h	AGPIO9_SerPortRX_OUT	GPIOx0A4	0000_0000h	Reserved
GPIOx028	0004_0000h	S0I3_GPIO _AGPIO10	GPIOx0A8	0004_0000h	S5_MUX_CTRL_EGPIO42
GPIOx02C	0014_0000h	BLINK_AGPIO11_USB_OC7_L _AGPIO11	GPIOx0AC	0000_0000h	Reserved
GPIOx030	0014_0000h	IR_LED_L_LLB_L_AGPIO12	GPIOx0B0	0000_0000h	Reserved
GPIOx034	0014_0000h	IR_TX0_USB_OC5_L _AGPIO13	GPIOx0B4	0000_0000h	Reserved
GPIOx038	0014_0000h	IR_TX1_USB_OC6_L_AGPIO 14	GPIOx0B8	0000_0000h	Int_FakeSts0
GPIOx03C	0014_0000h	IR_RX1_AGPIO15	GPIOx0BC	0000_0000h	Int_iEcSci
GPIOx040	0014_0000h	USB_OC0_L_TRST_L_AGPIO 16	GPIOx0C0	0000_0000h	Int_iCIR_Wake
GPIOx044	0014_0000h	USB_OC1_L_TDI_AGPIO17	GPIOx0C4	0000_0000h	Int_ASFSlaveIntr
GPIOx048	0014_0000h	USB_OC2_L_TCK_AGPIO18	GPIOx0C8	0000_0000h	Int_~ec_sm_irq_
GPIOx04C	0014_0000h	SCL1_I2C3_SCL_AGPIO19	GPIOx0CC	0000_0000h	Int_WakeFromLLB
GPIOx050	0014_0000h	SDA1_I2C3_SDA_AGPIO20	GPIOx0D0	0000_0000h	Int_AcDcTimerEvent
GPIOx054	0004_0000h	LPC_PD_L_AGPIO21	GPIOx0D4	0000_0000h	Int_ALTHPET_TimerSts
GPIOx058	0014_0000h	LPC_PME_L_AGPIO22	GPIOx0D8	0000_0000h	Int_iusb_wakeup0
GPIOx05C	0014_0000h	AC_PRES_USB_OC4_L_IR_RX0_AGPIO23	GPIOx0DC	0000_0000h	Int_iusb_wakeup1
GPIOx060	0014_0000h	TDO_USB_OC3_L_AGPIO24	GPIOx0E0	0000_0000h	Int_iusb_wakeup2
GPIOx064	0014_0000h	SD0_CD_AGPIO25	GPIOx0E4	0000_0000h	Int_iusb_wakeup3
GPIOx068	0004_0000h	PCIE_RST_L_EGPIO26	GPIOx0E8	0000_0000h	Int_usb_xhc_0_acpi_pme
GPIOx06C	0000_0000h	Reserved	GPIOx0EC	0000_0000h	Int_usb_xhc_1_acpi_pme
GPIOx070	0000_0000h	Reserved	GPIOx0F0	0000_0000h	Int_iAzPme
GPIOx074	0000_0000h	Reserved	GPIOx0F4	0000_0000h	Int_ACP_FCH_AZ_Wake
GPIOx078	0000_0000h	Reserved	GPIOx0F8	0000_0000h	Int_ ACP_FCH_I2S_Wake
GPIOx07C	0000_0000h	Reserved			

Table 279: Debounce Timer Definition

DebounceTmrLarge	DebounceTmrOutUnit	Timer Unit	Max Debounce Time
0	0	61 usec (2 RtcClk)	915 usec
0	1	183 usec (6 RtcClk)	2.75 msec
1	0	15.56 msec (510 RtcClk)	233 msec
1	1	62.44 msec (2046 RtcClk)	936 msec

External-GPIO register format. See Table 278 for GPIO function.

Bits	Description
31	**Less10secSts**. Read-only; Set-by-hardware. This bit is only valid for PWR_BTN_L_AGPIO0. For other GPIOs, this bit is Reserved. 1=The power button is pressed for less than 10 second in S0 state. This bit can be cleared by writing 1 to InterruptSts bit.
30	**Less2secSts**. Read-only; Set-by-hardware. This bit is only valid for PWR_BTN_L_AGPIO0. For other GPIOs, this bit is Reserved. 1=The power button is pressed for less than 2 second in S0 state. When Less2secSts becomes 1, Less10secSts will also become 1. This bit can be cleared by writing 1 to InterruptSts bit.
29	**WakeSts**. Read; Write-1-to-clear. 0=The pin doesn't generate wake event. 1=The pin is a wake source.
28	**InterruptSts**. Read; Write-1-to-clear. 0=The pin doesn't generate interrupt. 1=The pin is a interrupt source.
27:24	Reserved.
23	**OutputEnable**. Read-write. 0=Output is disabled on the pin. 1=Output is enabled on the pin.
22	**OutputValue**. Read-write. 0=Low. 1=High.
21	**PullDownEnable**. Read-write. 0=Pull-down is disabled on the pin. 1=Pull-down is enabled on the pin.
20	**PullUpEnable**. Read-write. 0=Pull-up is disabled on the pin. 1=Pull-up is enabled on the pin.
19:17	Reserved.
16	**PinSts**. Read-only; Updated-by-hardware. 0=The pin is low. 1=The pin is high. This bit is not affected by the debounce logic.
15:13	**WakeCntrl[2:0]**. Read-write. Bit — Definition [0] — 1=Enable wake in S0I3 state [1] — 1=Enable wake in S3 state (SLP_TYP = 3, and !G0_State) [2] — 1=Enable wake in S4/S5 state (SLP_TYP = 4 or 5, and !G0_State)
12:11	**InterruptEnable[1:0]**. Read-write. Bit — Definition [0] — 1=Enable interrupt status [1] — 1=Enable interrupt delivery
10:9	**ActiveLevel**. Read-write. Bits — Definition — Bits — Definition 00b — Active high — 10b — Active on both edges if LevelTrig == 0 01b — Active Low — 11b — Reserved
8	**LevelTrig**. Read-write. 0=Edge trigger. 1=Level trigger.

7	**DebounceTmrLarge**. Read-write. Combined with DebounceTmrOutUnit, this bit change the unit and max debounce time for the debounce timer. See DebounceTmrOutUnit and Table 279 [Debounce Timer Definition].
6:5	**DebounceCntrl**. Read-write. Bits Definition Bits Definition 00b No debounce 10b Preserve high glitch 01b Preserve low glitch 11b Remove glitch
4	**DebounceTmrOutUnit**. Read-write. DebounceTmrLarge and DebounceTmrOutUnit defines the unit and max debounce time for the debounce timer. See Table 279 [Debounce Timer Definition].
3:0	**DebounceTmrOut**. Read-write. Specifies the debounce timer out number. 0=Debounceing logic is disabled.

GPIOxFC GPIO_Wake_Inter_Master Switch

Bits	Description
31	**GpioWakeEn**. Read-write. Reset: 1. 0=All GPIO wake are blocked.
30	**GpioInterruptEn**. Read-write. Reset: 1. 0=All GPIO interrupts are blocked.
29	**EOI**. Read-write; Cleared-by-hardware. Reset: 1. This bit is set to 1 by software when it is allowed to send GPIO interrupt. Hardware clears this bit when interrupt occurs. When this bit is 0, the GPIO interrupt is blocked. This hardware and software handshake mechanism is the same with EOS of SMI.
28	**MaskStsEn**. Read-write. Reset: 1. 1=Enable hardware to block all wake/interrupt status generation when software writes to any debounce registers. The length of blocking depends on mask_sts_length[3:0].
27:24	**MaskStsLength[3:0]**. Read-write. Reset: 1111b. See MaskStsEn. The length of blocking = {MaskStsLength[11:0], 14'h3FFF}.
23:16	**MaskStsLength[11:4]**. Read-write. Reset: 0. See MaskStsEn and MaskStsLength[3:0].
15	**EnWinBlueBtn**. Read-write. Reset: 0. 0=GPIO0 detect debounced power button; Power button override is 4 seconds. 1=GPIO0 detect debounced power button in S3/S5/S0I3, and detect "pressed less than 2 seconds" and "pressed 2~10 seconds" in S0; Power button override is 10 seconds.
14	**IntrOutActiveHi**. Read-write. Reset: 0. 0=GPIO controller interrupt output is low active. 1=GPIO controller interrupt output is high active.
13	**SelGpio0Src:Select the source for GPIO0 detection**. Read-write. Reset: 0. 0=Power button goes to a processing logic first and then goes to GPIO0 detection logic. 1=Power button goes to GPIO0 debounce and then goes to GPIO0 detection logic. The “processing logic” includes 16ms debounce counter and a logic to detect how long the button has been pressed to generate press_less2s_sts and press_less4s_sts. “GPIO0 debounce block” only has debounce function.
12	**IntrOutPulse**. Read-write. Reset: 0. 0=GPIO controller interrupt output is a level signal. 1=GPIO controller interrupt output is pulse signal. The polarity is defined by IntrOutActiveHi register bit.
11:0	Reserved.

GPIOx[1FC:100:step4] GPIO Bank 1 Control Register

Each GPIO pin is controlled by 4 bytes. These registers control GPIO bank 1 pins: GPIO64-GPIO127.

Table 280: Reset Mapping for GPIOx[1FC:100:step4]

Register	Reset	Function	Register	Reset	Function
GPIOx100	0024_0000h	AGPIO64	GPIOx180	0020_0000h	SD0_CMD_EGPIO96
GPIOx104	0024_0000h	AGPIO65	GPIOx184	0020_0000h	SD0_DATA0_EGPIO97
GPIOx108	0024_0000h	AGPIO66_Shutdown_L	GPIOx188	0020_0000h	SD0_DATA1_EGPIO98
GPIOx10C	0004_0000h	EGPIO67_DEVSLP0	GPIOx18C	0020_0000h	SD0_DATA2_EGPIO99
GPIOx110	0024_0000h	AGPIO68_SGPIO_CLK	GPIOx190	0020_0000h	SD0_DATA3_EGPIO100
GPIOx114	0004_0000h	AGPIO69_SGPIO_LOAD	GPIOx194	0024_0000h	SD0_WP_EGPIO101
GPIOx118	0004_0000h	EGPIO70_DEVSLP1	GPIOx198	0024_0000h	SD0_PWR_CTRL_AGPIO102
GPIOx11C	0024_0000h	AGPIO71_SGPIO_DATAOUT	GPIOx19C	0000_0000h	Reserved
GPIOx120	0024_0000h	AGPIO72_SGPIO_DATAIN	GPIOx1A0	0000_0000h	Reserved
GPIOx124	0000_0000h	Reserved	GPIOx1A4	0000_0000h	Reserved
GPIOx128	0004_0000h	LPCCLK0_EGPIO74	GPIOx1A8	0000_0000h	Reserved
GPIOx12C	0004_0000h	LPCCLK1_EGPIO75	GPIOx1AC	0000_0000h	Reserved
GPIOx130	0004_0000h	AGPIO76_SPI_TPM_CS_L	GPIOx1B0	0000_0000h	Reserved
GPIOx134	0000_0000h	Reserved	GPIOx1B4	0000_0000h	Reserved
GPIOx138	0000_0000h	Reserved	GPIOx1B8	0000_0000h	Reserved
GPIOx13C	0000_0000h	Reserved	GPIOx1BC	0000_0000h	Reserved
GPIOx140	0000_0000h	Reserved	GPIOx1C0	0000_0000h	Reserved
GPIOx144	0000_0000h	Reserved	GPIOx1C4	0014_0000h	SCL0_I2C2_SCL_EGPIO113
GPIOx148	0000_0000h	Reserved	GPIOx1C8	0014_0000h	SDA0_I2C2_SDA_EGPIO114
GPIOx14C	0000_0000h	Reserved	GPIOx1CC	0014_0000h	CLK_REQ1_L_AGPIO115
GPIOx150	0014_0000h	FANIN0_AGPIO84	GPIOx1D0	0014_0000h	CLK_REQ2_L_AGPIO116
GPIOx154	0014_0000h	FANOUT0_AGPIO85	GPIOx1D4	0024_0000h	XHC_SPI_CLK_SPI_CLK_ESPI_CLK_EGPIO117
GPIOx158	0014_0000h	LPC_SMI_L_AGPIO86	GPIOx1D8	0004_0000h	XHC_SPI_CS1_L_SPI_CS1_L_EGPIO118
GPIOx15C	0014_0000h	SERIRQ_AGPIO87	GPIOx1DC	0004_0000h	SPI_CS2_L_ESPI_CS_L_EGPIO119
GPIOx160	0014_0000h	LPC_CLKRUN_L_AGPIO88	GPIOx1E0	0024_0000h	XHC_SPI_DI_SPI_DI_ESPI_DAT1_EGPIO120
GPIOx164	0014_0000h	GENINT1_L_AGPIO89	GPIOx1E4	0024_0000h	XHC_SPI_DO_SPI_DO_ESPI_DAT0_EGPIO121
GPIOx168	0014_0000h	GENINT2_L_AGPIO90	GPIOx1E8	0014_0000h	SPI_WP_L_ESPI_DAT2_EGPIO122
GPIOx16C	0024_0000h	SPKR_AGPIO91	GPIOx1EC	0000_0000h	Reserved
GPIOx170	0004_0000h	CLK_REQ0_L_SATA_IS0_L_SATA_ZP0_L_AGPIO92	GPIOx1F0	0000_0000h	Reserved
GPIOx174	0024_0000h	SD0_LED_EGPIO93	GPIOx1F4	0000_0000h	Reserved
GPIOx178	0000_0000h	Reserved	GPIOx1F8	0014_0000h	GA20IN_AGPIO126
GPIOx17C	0020_0000h	SD0_CLK_EGPIO95			

Bits	Description
31:30	Reserved.

29	**WakeSts**. Read; Write-1-to-clear. 0=The pin doesn't generate wake event. 1=The pin is a wake source.
28	**InterruptSts**. Read; Write-1-to-clear. 0=The pin doesn't generate interrupt. 1=The pin is a interrupt source.
27:24	Reserved.
23	**OutputEnable**. Read-write. 0=Output is disabled on the pin. 1=Output is enabled on the pin.
22	**OutputValue**. Read-write. 0=Low. 1=High.
21	**PullDownEnable**. Read-write. 0=Pull-down is disabled on the pin. 1=Pull-down is enabled on the pin.
20	**PullUpEnable**. Read-write. 0=Pull-up is disabled on the pin. 1=Pull-up is enabled on the pin.
19:17	Reserved.
16	**PinSts**. Read-only; Updated-by-hardware. 0=The pin is low. 1=The pin is high. This bit is not affected by the debounce logic.
15:13	**WakeCntrl[2:0]**. Read-write. Bit: Definition [0]: 1=Enable wake in S0I3 state [1]: 1=Enable wake in S3 state (SLP_TYP = 3, and !G0_State) [2]: 1=Enable wake in S4/S5 state (SLP_TYP = 4 or 5, and !G0_State)
12:11	**InterruptEnable[1:0]**. Read-write. Bit: Definition [0]: 1=Enable interrupt status [1]: 1=Enable interrupt delivery
10:9	**ActiveLevel**. Read-write. Bits: Definition — Bits: Definition 00b: Active high — 10b: Active on both edges if LevelTrig == 0 01b: Active Low — 11b: Reserved
8	**LevelTrig**. Read-write. 0=Edge trigger. 1=Level trigger.
7	**DebounceTmrLarge**. Read-write. Combined with DebounceTmrOutUnit, this bit change the unit and max debounce time for the debounce timer. See DebounceTmrOutUnit and Table 279 [Debounce Timer Definition].
6:5	**DebounceCntrl**. Read-write. Bits: Definition — Bits: Definition 00b: No debounce — 10b: Preserve high glitch 01b: Preserve low glitch — 11b: Remove glitch
4	**DebounceTmrOutUnit**. Read-write. DebounceTmrLarge and DebounceTmrOutUnit defines the unit and max debounce time for the debounce timer. See Table 279 [Debounce Timer Definition].
3:0	**DebounceTmrOut**. Read-write. Specifies the debounce timer out number. 0=Debounceing logic is disabled.

GPIOx[2DC:200:step4] GPIO Bank 2 Control Register

Each GPIO pin is controlled by 4 bytes. These registers control GPIO Bank 2 pins: GPIO128-GPIO183.

Table 281: Reset Mapping for GPIOx[2DC:200:step4]

Register	Reset	Function	Register	Reset	Function
GPIOx200	0000_0000h	Reserved	GPIOx280	0000_0000h	Reserved
GPIOx204	0014_0000h	KBRST_L_AGPIO129	GPIOx284	0000_0000h	Reserved
GPIOx208	0004_0000h	SATA_ACT_L_AGPIO130	GPIOx288	0000_0000h	Reserved
GPIOx20C	0004_0000h	CLK_REQ3_L_SATA_IS1_L_SATA_ZP1_L_EGPIO131	GPIOx28C	0000_0000h	Reserved
GPIOx210	0004_0000h	CLK_REQG_L_OSCIN_EGPIO132	GPIOx290	0000_0000h	Reserved
GPIOx214	0014_0000h	SPI_HOLD_L_ESPI_DAT3_EGPIO133	GPIOx294	0000_0000h	Reserved
GPIOx218	0000_0000h	Reserved	GPIOx298	0000_0000h	Reserved
GPIOx21C	0024_0000h	UART0_CTS_L_EGPIO135	GPIOx29C	0000_0000h	Reserved
GPIOx220	0024_0000h	UART0_RXD_EGPIO136	GPIOx2A0	0000_0000h	Reserved
GPIOx224	0014_0000h	UART0_RTS_L_EGPIO137	GPIOx2A4	0000_0000h	Reserved
GPIOx228	0014_0000h	UART0_TXD_EGPIO138	GPIOx2A8	0000_0000h	Reserved
GPIOx22C	0024_0000h	UART0_INTR_AGPIO139	GPIOx2AC	0000_0000h	Reserved
GPIOx230	0024_0000h	UART1_CTS_L_EGPIO140	GPIOx2B0	0000_0000h	Int_any_SBGppPme_any_SBGppHp
GPIOx234	0024_0000h	UART1_RXD_EGPIO141	GPIOx2B4	0000_0000h	Int_NBGppPmePulse_NBGppHpPulse
GPIOx238	0014_0000h	UART1_RTS_L_EGPIO142	GPIOx2B8	0000_0000h	Int_AcpiPerfIntr
GPIOx23C	0014_0000h	UART1_TXD_EGPIO143	GPIOx2BC	0000_0000h	Int_curFanStatus0_curTempStatus5
GPIOx240	0024_0000h	UART1_INTR_AGPIO144	GPIOx2C0	0000_0000h	Int_FanThermal_SCIOut
GPIOx244	0004_0000h	I2C0_SCL_EGPIO145	GPIOx2C4	0000_0000h	Int_ASFMasterIntr
GPIOx248	0004_0000h	I2C0_SDA_EGPIO146	GPIOx2C8	0000_0000h	Int_Ras_event
GPIOx24C	0004_0000h	I2C1_SCL_EGPIO147	GPIOx2CC	0000_0000h	Int_GBL_RLS
GPIOx250	0004_0000h	I2C1_SDA_EGPIO148	GPIOx2D0	0000_0000h	Int_ShortTimerEvent_LongTimerEvent
GPIOx254	0000_0000h	Reserved	GPIOx2D4	0000_0000h	Int_NBHwAssertion_r[3]_NBSciAssertion_r[3]
GPIOx258	0000_0000h	Reserved	GPIOx2D8	0000_0000h	eSPI_WAKE_PME_B
GPIOx25C	0000_0000h	Reserved	GPIOx2DC	0000_0000h	eSPI_SYS_EVT_B
GPIOx260	0000_0000h	Reserved			
GPIOx264	0000_0000h	Reserved			
GPIOx268	0000_0000h	Reserved			
GPIOx26C	0000_0000h	Reserved			
GPIOx270	0000_0000h	Reserved			
GPIOx274	0000_0000h	Reserved			
GPIOx278	0000_0000h	Reserved			
GPIOx27C	0000_0000h	Reserved			

Bits	Description
31:30	Reserved.
29	**WakeSts**. Read; Write-1-to-clear. 0=The pin doesn't generate wake event. 1=The pin is a wake source.
28	**InterruptSts**. Read; Write-1-to-clear. 0=The pin doesn't generate interrupt. 1=The pin is a interrupt source.
27:24	Reserved.
23	**OutputEnable**. Read-write. 0=Output is disabled on the pin. 1=Output is enabled on the pin.
22	**OutputValue**. Read-write. 0=Low. 1=High.
21	**PullDownEnable**. Read-write. 0=Pull-down is disabled on the pin. 1=Pull-down is enabled on the pin.
20	**PullUpEnable**. Read-write. 0=Pull-up is disabled on the pin. 1=Pull-up is enabled on the pin.
19:17	Reserved.
16	**PinSts**. Read-only; Updated-by-hardware. 0=The pin is low. 1=The pin is high. This bit is not affected by the debounce logic.
15:13	**WakeCntrl[2:0]**. Read-write. Bit — Definition [0] — 1=Enable wake in S0I3 state [1] — 1=Enable wake in S3 state (SLP_TYP = 3, and !G0_State) [2] — 1=Enable wake in S4/S5 state (SLP_TYP = 4 or 5, and !G0_State)
12:11	**InterruptEnable[1:0]**. Read-write. Bit — Definition [0] — 1=Enable interrupt status [1] — 1=Enable interrupt delivery
10:9	**ActiveLevel**. Read-write. Bits — Definition — Bits — Definition 00b — Active high — 10b — Active on both edges if LevelTrig == 0 01b — Active Low — 11b — Reserved
8	**LevelTrig**. Read-write. 0=Edge trigger. 1=Level trigger.
7	**DebounceTmrLarge**. Read-write. Combined with DebounceTmrOutUnit, this bit change the unit and max debounce time for the debounce timer. See DebounceTmrOutUnit and Table 279 [Debounce Timer Definition].
6:5	**DebounceCntrl**. Read-write. Bits — Definition — Bits — Definition 00b — No debounce — 10b — Preserve high glitch 01b — Preserve low glitch — 11b — Remove glitch
4	**DebounceTmrOutUnit**. Read-write. DebounceTmrLarge and DebounceTmrOutUnit defines the unit and max debounce time for the debounce timer. See Table 279 [Debounce Timer Definition].
3:0	**DebounceTmrOut**. Read-write. Specifies the debounce timer out number. 0=Debounceing logic is disabled.

GPIOx2F0 GPIO_Wake_Status_Index_0

Bits	Description
31:16	**WakeStatusIndex[31:16]**. Read-write. Reset: 0. For each bit, 1=At least one of the wake status of GPIO N*4 ~ N*4+3 is 1. Same as WakeStatusIndex[14:0].
15	Reserved.
14:0	**WakeStatusIndex[14:0]**. Read-write. Reset: 0. For each bit, 1=At least one of the wake status of GPIO N*4 ~ N*4+3 is 1. Take [3:0] as example: Bits Definition Bits Definition 0 any of GPIO [3:0] has Wake status 2 any of GPIO [11:8] has Wake status 1 any of GPIO [7:4] has Wake status 3 any of GPIO [15:12] has Wake status

GPIOx2F4 GPIO_Wake_Status_Index_1

Bits	Description
31:14	Reserved.
13:0	**WakeStatusIndex[13:0]**. Read-write. Reset: 0. For each bit, 1=At least one of the wake status of GPIO N*4 ~ N*4+3 is 1.

GPIOx2F8 GPIO_Interrupt_Status_Index_0

Bits	Description
31:16	**InterruptStatusIndex[31:16]**. Read-write. Reset: 0. For each bit, 1=At least one of the interrupt status of GPIO N*4 ~ N*4+3 is 1.
15	Reserved.
14:0	**InterruptStatusIndex[14:0]**. Read-write. Reset: 0. For each bit, 1=At least one of the interrupt status of GPIO N*4 ~ N*4+3 is 1.

GPIOx2FC GPIO_Interrupt_Status_Index_1

Bits	Description
31:29	Reserved.
28	**MaskStsEn**. Read-write. Reset: 1. 1=Enable hardware to block all wake/interrupt status generation when software writes to any debounce registers. The length of blocking depends on mask_sts_length[3:0].
27:24	**MaskStsLength[3:0]**. Read-write. Reset: 1111b. See MaskStsEn. The length of blocking = {MaskStsLength[11:0], 14'h3FFF}.
23:16	**MaskStsLength[11:4]**. Read-write. Reset: 0. See MaskStsEn and MaskStsLength[3:0].
15:14	Reserved.
13:0	**InterruptStatusIndex[13:0]**. Read-write. Reset: 0. For each bit, 1=At least one of the interrupt status of GPIO N*4 ~ N*4+3 is 1.

3.26.11.2 IOMux Registers

IOMux register space is accessed through the AcpiMmio region. The registers range from FED8_0000h+D00h to FED8_0000h+DFFh. See PMx04[Mmio].

IOMUXx00 PWR_BTN_L_AGPIO0

Bits	Description
7:2	Reserved.
1:0	**PWR_BTN_L_AGPIO0**. Read-write. Reset: 00b. Multi-function IO pin function select for pin PWR_BTN_L_AGPIO0. Bits: Definition 00b: PWR_BTN_L 01b: AGPIO0 10b: Reserved 11b: Reserved

IOMUXx01 SYS_RESET_L_AGPIO1

Bits	Description
7:2	Reserved.
1:0	**SYS_RESET_L_AGPIO1**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SYS_RESET_L_AGPIO1. Bits: Definition 00b: SYS_RESET_L 01b: AGPIO1 10b: Reserved 11b: Reserved

IOMUXx02 WAKE_L_AGPIO2

Bits	Description
7:2	Reserved.
1:0	**WAKE_L_AGPIO2**. Read-write. Reset: 00b. Multi-function IO pin function select for pin WAKE_L_AGPIO2. Bits: Definition 00b: WAKE_L 01b: AGPIO2 10b: Reserved 11b: Reserved

IOMUXx03 AGPIO3

Bits	Description
7:2	Reserved.
1:0	**AGPIO3**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO3. See 2.17.4.3 to use AGPIO3 as LPC_SMI function. Bits Definition 00b AGPIO3 01b Reserved 10b Reserved 11b Reserved

IOMUXx04 AGPIO4

Bits	Description
7:2	Reserved.
1:0	**AGPIO4**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO4. Bits Definition 00b AGPIO4 01b Reserved 10b Reserved 11b Reserved

IOMUXx05 AGPIO5_DEVSLP0_S5

Bits	Description
7:2	Reserved.
1:0	**AGPIO5_DEVSLP0_S5**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO5_DEVSLP0_S5. Bits Definition 00b AGPIO5 01b DEVSLP0_S5 10b Reserved 11b Reserved

IOMUXx06 AGPIO6_LDT_RST_L

Bits	Description
7:2	Reserved.
1:0	**AGPIO6_LDT_RST_L**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO6_LDT_RST_L. See 2.17.4.3 to use AGPIO6 as LPC_SMI function. Bits Definition 00b AGPIO6 01b LDT_RST_L 10b Reserved 11b Reserved

IOMUXx07 AGPIO7_LDT_PWROK

Bits	Description
7:2	Reserved.
1:0	**AGPIO7_LDT_PWROK**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO7_LDT_PWROK. Bits Definition 00b AGPIO7 01b LDT_PWROK 10b Reserved 11b Reserved

IOMUXx08 AGPIO8_SerPortTX_OUT

Bits	Description
7:2	Reserved.
1:0	**AGPIO8_SerPortTX_OUT**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO8_SerPortTX_OUT. Bits Definition 00b AGPIO8 01b SerPortTX_OUT 10b Reserved 11b Reserved

IOMUXx09 AGPIO9_SerPortRX_OUT

Bits	Description
7:2	Reserved.
1:0	**AGPIO9_SerPortRX_OUT**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO9_SerPortRX_OUT. Bits Definition 00b AGPIO9 01b SerPortRX_OUT 10b Reserved 11b Reserved

IOMUXx0A S0I3_GPIO_AGPIO10

Bits	Description
7:2	Reserved.
1:0	**S0I3_GPIO_AGPIO10**. Read-write. Reset: 00b. Multi-function IO pin function select for pin S0I3_GPIO_AGPIO10. Bits Definition 00b S0I3 01b AGPIO10 10b Reserved 11b Reserved

IOMUXx0B BLINK _AGPIO11_USB_OC7_L_AGPIO11

Bits	Description
7:2	Reserved.
1:0	**BLINK_AGPIO11_USB_OC7_L_AGPIO11**. Read-write. Reset: 00b. Multi-function IO pin function select for pin BLINK. If overrideen, then the IO pin uses the BLINK function and the GPIO function is disabled. Bits Definition 00b AGPIO11 01b USB_OC7_L 10b AGPIO11 11b Reserved

IOMUXx0C IR_LED_L_LLB_L_AGPIO12

Bits	Description
7:2	Reserved.
1:0	**IR_LED_L_LLB_L_AGPIO12**. Read-write. Reset: 00b. Multi-function IO pin function select for pin IR_LED_L_LLB_L_AGPIO12. Bits Definition 00b IR_LED_L 01b LLB_L 10b AGPIO12 11b Reserved

IOMUXx0D IR_TX0_USB_OC5_L_AGPIO13

Bits	Description
7:2	Reserved.
1:0	**IR_TX0_USB_OC5_L_AGPIO13**. Read-write. Reset: 00b. Multi-function IO pin function select for pin IR_TX0_USB_OC5_L_AGPIO13. Bits Definition 00b USB_OC5_L 01b AGPIO13 10b Reserved 11b Reserved

IOMUXx0E IR_TX1_USB_OC6_L_AGPIO14

Bits	Description
7:2	Reserved.
1:0	**IR_TX1_USB_OC6_L_AGPIO14**. Read-write. Reset: 00b. Multi-function IO pin function select for pin IR_TX1_USB_OC6_L_AGPIO14. Bits Definition 00b USB_OC6_L 01b AGPIO14 10b Reserved 11b Reserved

IOMUXx0F IR_RX1_AGPIO15

Bits	Description
7:2	Reserved.
1:0	**IR_RX1_AGPIO15**. Read-write. Reset: 00b. Multi-function IO pin function select for pin IR_RX1_AGPIO15. See 2.17.4.3 to use AGPIO15 as LPC_SMI function. Bits Definition 00b IR_RX1 01b AGPIO15 10b Reserved 11b Reserved

IOMUXx10 USB_OC0_L_TRST_L_AGPIO16

Bits	Description
7:2	Reserved.
1:0	**USB_OC0_L_TRST_L_AGPIO16**. Read-write. Reset: 00b. Multi-function IO pin function select for pin USB_OC0_L_TRST_L_AGPIO16. Bits Definition 00b USB_OC0_L 01b TRST_L 10b AGPIO16 11b Reserved

IOMUXx11 USB_OC1_L_TDI_AGPIO17

Bits	Description
7:2	Reserved.
1:0	**USB_OC1_L_TDI_AGPIO17**. Read-write. Reset: 00b. Multi-function IO pin function select for pin USB_OC1_L_TDI_AGPIO17. Bits Definition 00b USB_OC1_L 01b TDI 10b AGPIO17 11b Reserved

IOMUXx12 USB_OC2_L_TCK_AGPIO18

Bits	Description
7:2	Reserved.
1:0	**USB_OC2_L_TCK_AGPIO18**. Read-write. Reset: 00b. Multi-function IO pin function select for pin USB_OC2_L_TCK_AGPIO18. Bits Definition 00b USB_OC2_L 01b TCK 10b AGPIO18 11b Reserved

IOMUXx13 SCL1_I2C3_SCL_AGPIO19

Bits	Description
7:2	Reserved.
1:0	**SCL1_I2C3_SCL_AGPIO19**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SCL1_I2C3_SCL_EGPIO19. Bits Definition 00b SCL1 01b I2C3_SCL 10b AGPIO19 11b Reserved

IOMUXx14 SDA1_I2C3_SDA_AGPIO20

Bits	Description
7:2	Reserved.
1:0	**SDA1_I2C3_SDA_AGPIO20**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SDA1_I2C3_SDA_EGPIO20. Bits Definition 00b SDA1 01b I2C3_SDA 10b AGPIO20 11b Reserved

IOMUXx15 LPC_PD_L_AGPIO21

Bits	Description
7:2	Reserved.
1:0	**LPC_PD_L_AGPIO21**. Read-write. Reset: 00b. Multi-function IO pin function select for pin LPC_PD_L_AGPIO21. See 2.17.4.3 to use AGPIO21 as LPC_SMI function. Bits Definition 00b LPC_PD_L 01b AGPIO21 10b Reserved 11b Reserved

IOMUXx16 LPC_PME_L_AGPIO22

Bits	Description
7:2	Reserved.
1:0	**LPC_PME_L_AGPIO22**. Read-write. Reset: 00b. Multi-function IO pin function select for pin LPC_PME_L_AGPIO22. See 2.17.4.3 to use AGPIO22 as LPC_SMI function. Bits Definition 00b LPC_PME_L 01b AGPIO22 10b Reserved 11b Reserved

IOMUXx17 AC_PRES_USB_OC4_L_IR_RX0_AGPIO23

Bits	Description
7:2	Reserved.
1:0	**AC_PRES_USB_OC4_L_IR_RX0_AGPIO23**. Read-write. Reset: 01b. Multi-function IO pin function select for pin AC_PRES_USB_OC4_L_IR_RX0_AGPIO23. Bits Definition 00b USB_OC4_L 01b IR_RX0 10b AGPIO23 11b Reserved Program PMxA0[BatteryModeEn] = 1 to enable AC_PRES as defined by IOMUX control.

IOMUXx18 TDO_USB_OC3_L_AGPIO24

Bits	Description
7:2	Reserved.
1:0	**TDO_USB_OC3_L_AGPIO24**. Read-write. Reset: 00b. Multi-function IO pin function select for pin USB_OC3_L_AGPIO24. See 2.17.4.3 to use AGPIO24 as LPC_SMI function. Bits Definition 00b USB_OC3_L 01b AGPIO24 10b Reserved 11b Reserved

IOMUXx19 SD0_CD_AGPIO25

Bits	Description
7:2	Reserved.
1:0	**SD0_CD_AGPIO25**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SD0_CD_AGPIO25. Bits Definition 00b SD0_CD 01b AGPIO25 10b Reserved 11b Reserved

IOMUXx1A PCIE_RST_L_EGPIO26

Bits	Description
7:2	Reserved.
1:0	**PCIE_RST_L_EGPIO26**. Read-write. Reset: 00b. Multi-function IO pin function select for pin PCIE_RST_L_EGPIO26. Bits Definition 00b PCIE_RST_L 01b EGPIO26 10b Reserved 11b Reserved

IOMUXx27 VDDGFX_PD_AGPIO39

Bits	Description
7:2	Reserved.
1:0	**VDDGFX_PD_AGPIO39**. Read-write. Reset: 00b. Multi-function IO pin function select for pin VDDGFX_PD_AGPIO39. Bits Definition 00b VDDGFX_PD as a input pin 01b AGPIO39 as a bi-direction GPIO 10b Reserved 11b Reserved

IOMUXx28 AGPIO40

Bits	Description
7:2	Reserved.
1:0	**AGPIO40**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO40. Bits Definition 00b AGPIO40 01b Reserved 10b Reserved 11b Reserved

IOMUXx2A S5_MUX_CTRL_EGPIO42

Bits	Description
7:2	Reserved.
1:0	**S5_MUX_CTRL_EGPIO42**. Read-write. Reset: 00b. Multi-function IO pin function select for pin S5_MUX_CTRL_EGPIO42. Bits Definition 00b S5_MUX_CTRL 01b EGPIO42 10b Reserved 11b Reserved

IOMUXx40 AGPIO64

Bits	Description
7:2	Reserved.
1:0	**AGPIO64**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO64. Bits Definition 00b AGPIO64 01b Reserved 10b Reserved 11b Reserved

IOMUXx41 AGPIO65

Bits	Description
7:2	Reserved.
1:0	**AGPIO65**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO65. Bits Definition 00b AGPIO65 01b Reserved 10b Reserved 11b Reserved

IOMUXx42 AGPIO66_ShutDown_L

Bits	Description
7:2	Reserved.
1:0	**AGPIO66_ShutDown_L**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO66_Shutdown_L. Bits Definition 00b AGPIO66 01b Shutdown_L 10b Reserved 11b Reserved

IOMUXx43 EGPIO67_DEVSLP0

Bits	Description
7:2	Reserved.
1:0	**EGPIO67_DEVSLP0**. Read-write. Reset: 00b. Multi-function IO pin function select for pin EGPIO67_DEVSLP0. Bits Definition 00b EGPIO67 01b DEVSLP0 10b Reserved 11b Reserved

IOMUXx44 AGPIO68_SGPIO_CLK

Bits	Description
7:2	Reserved.
1:0	**AGPIO68_SGPIO_CLK**. Read-write. Reset: 00b. Multi-function IO pin function select for AGPIO68_SGPIO_CLK. Bits Definition 00b AGPIO68 01b SGPIO_CLK 10b Reserved 11b Reserved

IOMUXx45 AGPIO69_SGPIO_LOAD

Bits	Description
7:2	Reserved.
1:0	**AGPIO69_SGPIO_LOAD**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO69_SGPIO_LOAD. Bits Definition 00b AGPIO69 01b SGPIO_LOAD 10b Reserved 11b Reserved

IOMUXx46 EGPIO70_DEVSLP1

Bits	Description
7:2	Reserved.
1:0	**EGPIO70_DEVSLP1**. Read-write. Reset: 00b. Multi-function IO pin function select for EGPIO70_DEVSLP1. Bits Definition 00b EGPIO70 01b DEVSLP1 10b Reserved 11b Reserved

IOMUXx47 AGPIO71_SGPIO_DATAOUT

Bits	Description
7:2	Reserved.
1:0	**AGPIO71_SGPIO_DATAOUT**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO71_SGPIO_DATAOUT. Bits Definition 00b AGPIO71 01b SGPIO_DATAOUT 10b Reserved 11b Reserved

IOMUXx48 AGPIO72_SGPIO_DATAIN

Bits	Description
7:2	Reserved.
1:0	**AGPIO72_SGPIO_DATAIN**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO72_SGPIO_DATAIN. Bits Definition 00b AGPIO72 01b SGPIO_DATAIN 10b Reserved 11b Reserved

IOMUXx4A LPCCLK0_EGPIO74

Bits	Description
7:2	Reserved.
1:0	**LPCCLK0_EGPIO74**. Read-write. Reset: 00b. Multi-function IO pin function select for pin LPCCLK0_EGPIO74. Bits Definition 00b LPCCLK0 01b EGPIO74 10b Reserved 11b Reserved

IOMUXx4B LPCCLK1_EGPIO75

Bits	Description
7:2	Reserved.
1:0	**LPCCLK1_EGPIO75**. Read-write. Reset: 00b. Multi-function IO pin function select for pin LPCCLK1_EGPIO75. Bits Definition 00b LPCCLK1 01b EGPIO75 10b Reserved 11b Reserved

IOMUXx4C AGPIO76_SPI_TPM_CS_L

Bits	Description
7:2	Reserved.
1:0	**AGPIO76_SPI_TPM_CS_L**. Read-write. Reset: 00b. Multi-function IO pin function select for pin AGPIO76_SPI_TPM_CS_L. Bits Definition 00b AGPIO76 01b SPI_TPM_CS_L 10b Reserved 11b Reserved

IOMUXx54 FANIN0_AGPIO84

Bits	Description
7:2	Reserved.
1:0	**FANIN0_AGPIO84**. Read-write. Reset: 00b. Multi-function IO pin function select for pin FANIN0_AGPIO84. Bits Definition 00b FANIN0 01b AGPIO84 10b Reserved 11b Reserved

IOMUXx55 FANOUT0_AGPIO85

Bits	Description
7:2	Reserved.
1:0	**FANOUT0_AGPIO85**. Read-write. Reset: 01b. Multi-function IO pin function select for pin FANOUT0_AGPIO85. Bits Definition 00b FANOUT0 01b AGPIO85 10b Reserved 11b Reserved

IOMUXx56 LPC_SMI_L_AGPIO86

Bits	Description
7:2	Reserved.
1:0	**LPC_SMI_L_AGPIO86**. Read-write. Reset: 00b. Multi-function IO pin function select for pin LPC_SMI_L_AGPIO86. See 2.17.4.3 to use AGPIO21 as LPC_SMI function. Bits Definition 00b Reserved 01b AGPIO86 10b Reserved 11b Reserved

IOMUXx57 SERIRQ_AGPIO87

Bits	Description
7:2	Reserved.
1:0	**SERIRQ_AGPIO87**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SERIRQ_AGPIO87. Bits Definition 00b SERIRQ 01b AGPIO87 10b Reserved 11b Reserved

IOMUXx58 LPC_CLKRUN_L_AGPIO88

Bits	Description
7:2	Reserved.
1:0	**LPC_CLKRUN_L_AGPIO88**. Read-write. Reset: 00b. Multi-function IO pin function select for LPC_CLKRUN_L_AGPIO88. Bits Definition 00b LPC_CLKRUN_L 01b AGPIO88 10b Reserved 11b Reserved

IOMUXx59 GENINT1_L_AGPIO89

Bits	Description
7:2	Reserved.
1:0	**GENINT1_L_AGPIO89**. Read-write. Reset: 00b. Multi-function IO pin configurable either as GENINT1_L or as AGPIO89. Bits Definition 00b AGPIO89 01b Reserved 10b Reserved 11b Reserved

IOMUXx5A GENINT2_L_AGPIO90

Bits	Description
7:2	Reserved.
1:0	**GENINT2_L_AGPIO90**. Read-write. Reset: 00b. Multi-function IO pin configurable either as GENINT2_L or as AGPIO90. Bits Definition 00b AGPIO90 01b Reserved 10b Reserved 11b Reserved

IOMUXx5B SPKR_AGPIO91

Bits	Description
7:2	Reserved.
1:0	**SPKR_AGPIO91**. Read-write. Reset: 01b. Multi-function IO pin function select for pin SPKR_AGPIO91. Bits Definition 00b SPKR 01b AGPIO91 10b Reserved 11b Reserved

IOMUXx5C CLK_REQ0_L_SATA_IS0_L_SATA_ZP0_L_AGPIO92

Bits	Description
7:2	Reserved.
1:0	**CLK_REQ0_L_SATA_IS0_L_SATA_ZP0_L_AGPIO92**. Read-write. Reset: 00b. Multi-function IO pin function select for pin CLK_REQ0_L_SATA_IS0_L_SATA_ZP0_L_AGPIO92. Bits Definition 00b CLK_REQ0_L 01b SATA_IS0_L 10b SATA_ZP0_L 11b AGPIO92

IOMUXx5D SD0_LED_EGPIO93

Bits	Description
7:2	Reserved.
1:0	**SD0_LED_EGPIO93**. Read-write. Reset: 01b. Multi-function IO pin function select for pin SD0_LED_EGPIO93. Bits Definition 00b SD0_LED 01b EGPIO93 10b Reserved 11b Reserved

IOMUXx5F SD0_CLK_EGPIO95

Bits	Description
7:2	Reserved.
1:0	**SD0_CLK_EGPIO95**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SD0_CLK_EGPIO95. Bits Definition 00b EGPIO95 01b Reserved 10b Reserved 11b Reserved

IOMUXx60 SD0_CMD_EGPIO96

Bits	Description
7:2	Reserved.
1:0	**SD0_CMD_EGPIO96**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SD0_CMD_EGPIO96. Bits Definition 00b EGPIO96 01b Reserved 10b Reserved 11b Reserved

IOMUXx[64:61] SD0_DATA[3:0]_EGPIO[100:97]

Bits	Description
7:2	Reserved.
1:0	**SD0_DATA[3:0]_EGPIO[100:97]**. Read-write. Reset: 00b. Multi-function IO pin function select for SD0_DATA[3:0]. Bits Definition 00b EGPIO[100:97] 01b Reserved 10b Reserved 11b Reserved

IOMUXx65 SD0_WP_EGPIO101

Bits	Description
7:2	Reserved.
1:0	**SD0_WP_EGPIO101**. Read-write. Reset: 00b. Multi-function IO pin function select for SD0_WP_EGPIO101. Bits Definition 00b SD0_WP 01b EGPIO101 10b Reserved 11b Reserved

IOMUXx66 SD0_PWR_CTRL_AGPIO102

Bits	Description
7:2	Reserved.
1:0	**SD0_PWR_CTRL_AGPIO102**. Read-write. Reset: 01b. Multi-function IO pin function select for SD0_PWR_CTRL_AGPIO102. Bits Definition 00b SD0_PWR_CTRL 01b AGPIO102 10b Reserved 11b Reserved

IOMUXx71 SCL0_I2C2_SCL_EGPIO113

Bits	Description
7:2	Reserved.
1:0	**SCL0_I2C2_SCL_EGPIO113**. Read-write. Reset: 00b. Multi-function IO pin function select for SCL0_I2C2_SCL_EGPIO113. Bits Definition 00b SCL0 01b I2C2_SCL 10b EGPIO113 11b Reserved

IOMUXx72 SDA0_I2C2_SDA_EGPIO114

Bits	Description
7:2	Reserved.
1:0	**SDA0_I2C2_SDA_EGPIO114**. Read-write. Reset: 00b. Multi-function IO pin function select for SDA0_I2C2_SDA_EGPIO114. Bits Definition 00b SDA0 01b I2C2_SDA 10b EGPIO114 11b Reserved

IOMUXx73 CLK_REQ1_L_AGPIO115

Bits	Description
7:2	Reserved.
1:0	**CLK_REQ1_L_AGPIO115**. Read-write. Reset: 00b. Multi-function IO pin function select for pin CLK_REQ1_L_AGPIO115. Bits Definition 00b CLK_REQ1_L 01b AGPIO115 10b Reserved 11b Reserved

IOMUXx74 CLK_REQ2_L_AGPIO116

Bits	Description
7:2	Reserved.
1:0	**CLK_REQ2_L_AGPIO116**. Read-write. Reset: 00b. Multi-function IO pin function select for pin CLK_REQ2_L_AGPIO116. Bits Definition 00b CLK_REQ2_L 01b AGPIO116 10b Reserved 11b Reserved

IOMUXx75 XHC_SPI_CLK_SPI_CLK_ESPI_CLK_EGPIO117

Bits	Description
7:2	Reserved.
1:0	**XHC_SPI_CLK_SPI_CLK_ESPI_CLK_EGPIO117**. Read-write. Reset: 00b. Multi-function IO pin function select for pin XHC_SPI_CLK_SPI_CLK_ESPI_CLK_EGPIO117. Bits Definition 00b ESPI_CLK 01b EGPIO117 10b Reserved 11b Reserved

IOMUXx76 XHC_SPI_CS1_L_SPI_CS1_L_EGPIO118

Bits	Description
7:2	Reserved.
1:0	**XHC_SPI_CS1_L_SPI_CS1_L_EGPIO118**. Read-write. Reset: 00b. Multi-function IO pin function select for pin XHC_SPI_CS1_L_SPI_CS1_L_EGPIO118. Bits Definition 00b SPI_CS1_L 01b EGPIO118 10b Reserved 11b Reserved

IOMUXx77 SPI_CS2_L_ESPI_CS_L_EGPIO119

Bits	Description
7:2	Reserved.
1:0	**SPI_CS2_L_ESPI_CS_L_EGPIO119**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SPI_CS2_L_ESPI_CS_L_EGPIO119. Bits Definition 00b SPI_CS2_L 01b ESPI_CS_L 10b EGPIO119 11b Reserved

IOMUXx78 XHC_SPI_DI_SPI_DI_ESPI_DAT1_EGPIO120

Bits	Description
7:2	Reserved.
1:0	**XHC_SPI_DI_SPI_DI_ESPI_DAT1_EGPIO120**. Read-write. Reset: 00b. Multi-function IO pin function select for pin XHC_SPI_DI_SPI_DI_ESPI_DAT1_EGPIO120. Bits Definition 00b ESPI_DAT1 01b EGPIO120 10b Reserved 11b Reserved

IOMUXx79 XHC_SPI_DO_SPI_DO_ESPI_DAT0_EGPIO121

Bits	Description
7:2	Reserved.
1:0	**XHC_SPI_DO_SPI_DO_ESPI_DAT0_EGPIO121**. Read-write. Reset: 00b. Multi-function IO pin function select for pin XHC_SPI_DO_SPI_DO_ESPI_DAT0_EGPIO121. Bits Definition 00b ESPI_DAT0 01b EGPIO121 10b Reserved 11b Reserved

IOMUXx7A SPI_WP_L_ESPI_DAT2_EGPIO122

Bits	Description
7:2	Reserved.
1:0	**SPI_WP_L_ESPI_DAT2_EGPIO122**. Read-write. Reset: 01b. Multi-function IO pin function select for pin SPI_WP_L_ESPI_DAT2_EGPIO122. Bits Definition 00b ESPI_DAT2 01b EGPIO122 10b Reserved 11b Reserved

IOMUXx7E GA20IN_AGPIO126

Bits	Description
7:2	Reserved.
1:0	**GA20IN_AGPIO126**. Read-write. Reset: 00b. Multi-function IO pin function select for pin GA20IN_AGPIO126. Bits Definition 00b GA20IN 01b AGPIO126 10b Reserved 11b Reserved

IOMUXx81 KBRST_L_AGPIO129

Bits	Description
7:2	Reserved.
1:0	**KBRST_L_AGPIO129**. Read-write. Reset: 00b. Multi-function IO pin function select for pin KBRST_L_AGPIO129. Bits Definition 00b KBRST_L 01b AGPIO129 10b Reserved 11b Reserved

IOMUXx82 SATA_ACT_L_AGPIO130

Bits	Description
7:2	Reserved.
1:0	**SATA_ACT_L_AGPIO130**. Read-write. Reset: 00b. Multi-function IO pin function select for pin SATA_ACT_L_AGPIO130. Bits Definition 00b SATA_ACT_L 01b AGPIO130 10b Reserved 11b Reserved

IOMUXx83 CLK_REQ3_L_SATA_IS1_L_SATA_ZP1_L_EGPIO131

Bits	Description
7:2	Reserved.
1:0	**CLK_REQ3_L_SATA_IS1_L_SATA_ZP1_L_EGPIO131**. Read-write. Reset: 00b. Multi-function IO pin function select for pin CLK_REQ3_L_SATA_IS1_L_SATA_ZP1_L_EGPIO131. Bits Definition 00b CLK_REQ3_L 01b SATA_IS1_L 10b SATA_ZP1_L 11b EGPIO131

IOMUXx84 CLK_REQG_L_OSCIN_EGPIO132

Bits	Description
7:2	Reserved.
1:0	**CLK_REQG_L_OSCIN_EGPIO132**. Read-write. Reset: 00b. Multi-function IO pin function select for pin CLK_REQG_L_OSCIN_EGPIO132. Bits Definition 00b CLK_REQG_L 01b OSCIN 10b EGPIO132 11b Reserved

IOMUXx85 SPI_HOLD_L_ESPI_DAT3_EGPIO133

Bits	Description
7:2	Reserved.
1:0	**SPI_HOLD_L_ESPI_DAT3_EGPIO133**. Read-write. Reset: 01b. Multi-function IO pin function select for pin SPI_HOLD_L_ESPI_DAT3_EGPIO133. Bits Definition 00b ESPI_DAT3 01b EGPIO133 10b Reserved 11b Reserved

IOMUXx87 UART0_CTS_L_EGPIO135

Bits	Description
7:2	Reserved.
1:0	**UART0_CTS_L_EGPIO135**. Read-write. Reset: 00b. Multi-function IO pin function select for pin UART0_CTS_L_EGPIO135. Bits Definition 00b UART0_CTS_L 01b EGPIO135 10b Reserved 11b Reserved

IOMUXx88 UART0_RXD_EGPIO136

Bits	Description
7:2	Reserved.
1:0	**UART0_RXD_EGPIO136**. Read-write. Reset: 00b. Multi-function IO pin function select for pin CLK_REQG_EGPIO136. Bits Definition 00b UART0_RXD 01b EGPIO136 10b Reserved 11b Reserved

IOMUXx89 UART0_RTS_L_EGPIO137

Bits	Description
7:2	Reserved.
1:0	**UART0_RTS_L_EGPIO137**. Read-write. Reset: 01b. Multi-function IO pin function select for pin UART0_RTS_L_EGPIO137. Bits Definition 00b UART0_RTS_L 01b EGPIO137 10b Reserved 11b Reserved

IOMUXx8A UART0_TXD_EGPIO138

Bits	Description
7:2	Reserved.
1:0	**UART0_TXD_EGPIO138**. Read-write. Reset: 01b. Multi-function IO pin function select for pin UART0_TXD_EGPIO138. Bits Definition 00b UART0_TXD 01b EGPIO138 10b Reserved 11b Reserved

IOMUXx8B UART0_INTR_AGPIO139

Bits	Description
7:2	Reserved.
1:0	**UART0_INTR_AGPIO139**. Read-write. Reset: 00b. Multi-function IO pin function select for pin UART0_INTR_AGPIO139. Bits Definition 00b UART0_INTR 01b AGPIO139 10b Reserved 11b Reserved

IOMUXx8C UART1_CTS_L_EGPIO140

Bits	Description
7:2	Reserved.
1:0	**UART1_CTS_L_EGPIO140**. Read-write. Reset: 00b. Multi-function IO pin function select for pin UART1_CTS_L_EGPIO140. Bits Definition 00b UART1_CTS_L 01b EGPIO140 10b Reserved 11b Reserved

IOMUXx8D UART1_RXD_EGPIO141

Bits	Description
7:2	Reserved.
1:0	**UART1_RXD_EGPIO141**. Read-write. Reset: 00b. Multi-function IO pin function select for pin UART1_RXD_EGPIO141. Bits Definition 00b UART1_RXD 01b EGPIO141 10b Reserved 11b Reserved

IOMUXx8E UART1_RTS_L_EGPIO142

Bits	Description
7:2	Reserved.
1:0	**UART1_RTS_L_EGPIO142**. Read-write. Reset: 01b. Multi-function IO pin function select for pin UART1_RTS_L_EGPIO142. Bits Definition 00b UART1_RTS_L 01b EGPIO142 10b Reserved 11b Reserved

IOMUXx8F UART1_TXD_EGPIO143

Bits	Description
7:2	Reserved.
1:0	**UART1_TXD_EGPIO143**. Read-write. Reset: 01b. Multi-function IO pin function select for pin UART1_TXD_EGPIO143. Bits Definition 00b UART1_TXD 01b EGPIO143 10b Reserved 11b Reserved

IOMUXx90 UART1_INTR_AGPIO144

Bits	Description
7:2	Reserved.
1:0	**UART1_INTR_AGPIO144**. Read-write. Reset: 00b. Multi-function IO pin function select for pin UART1_INTR_AGPIO144. Bits Definition 00b UART1_INTR 01b AGPIO144 10b Reserved 11b Reserved

IOMUXx91 I2C0_SCL_EGPIO145

Bits	Description
7:2	Reserved.
1:0	**I2C0_SCL_EGPIO145**. Read-write. Reset: 00b. Multi-function IO pin function select for pin I2C0_SCL_EGPIO145. Bits Definition 00b I2C0_SCL 01b EGPIO145 10b Reserved 11b Reserved

IOMUXx92 I2C0_SDA_EGPIO146

Bits	Description
7:2	Reserved.
1:0	**I2C0_SDA_EGPIO146**. Read-write. Reset: 00b. Multi-function IO pin function select for pin I2C0_SDA_EGPIO146. Bits Definition 00b I2C0_SDA 01b EGPIO146 10b Reserved 11b Reserved

IOMUXx93 I2C1_SCL_EGPIO147

Bits	Description
7:2	Reserved.
1:0	**I2C1_SCL_EGPIO147**. Read-write. Reset: 00b. Multi-function IO pin function select for pin I2C1_SCL_EGPIO147. Bits Definition 00b I2C1_SCL 01b EGPIO147 10b Reserved 11b Reserved

IOMUXx94 I2C1_SDA_EGPIO148

Bits	Description
7:2	Reserved.
1:0	**I2C1_SDA_EGPIO148**. Read-write. Reset: 00b. Multi-function IO pin function select for pin I2C1_SDA_EGPIO148. Bits Definition 00b I2C1_SDA 01b EGPIO148 10b Reserved 11b Reserved

3.26.12 Power Management (PM) Registers

PM register space is accessed through two methods:
- Indirect IO access through IOCD6 [PM_Index] and IOCD7 [PM_Data]. Software first programs the offset into the index register IOCD6 and then reads from or writes to the data register IOCD7.
- Direct memory mapped access through the AcpiMmio region. The PM registers range from FED8_0000h+300h to FED8_0000h+3FFh. See PMx04[MmioEn] for details on the AcpiMmio region.

PMx00 DecodeEn

Bits	Description
31:30	Reserved.
29	**HpetMsiEn**. Read-write. Reset: 1. 1=Expose MSI capability in HPET Capbility register.
28	**HpetWidthSel**. Read-write. Reset: 0. 0=HPET is 32-bit. 1=HPET is 64-bit.
27:26	**WatchDogOptions**. Read-write. Reset: 00b. 00b=Enable WatchDogTimer.
25:24	**WatchDogFreq**. Read-write. Reset: 11b. Defines the clock frequency used by the WatchDogTimer. Bits: Definition: Bits: Definition 00b: 32 us: 10b: 100 ms 01b: 10 ms: 11b: 1 s
23:21	**AsfClkSel**. Read-write. Reset: 000b. Specifies the frequency of ASF master clock. Bits: Definition: Bits: Definition 0h: ~100 kHz: 4h: ~600 kHz 1h: ~200 kHz: 5h: ~800 kHz 2h: ~300 kHz: 6h: ~900 kHz 3h: ~400 kHz: 7h: ~1 MHz
20:19	**SmBus0Sel**. Read-write. Reset: 0. Specifies SMBus port selection. There is only one SMBus engine controlling four SMBus ports. This register routes the SMBus engine to the desired port. Bits: Definition: Bits: Definition 00b: Port 0, for SMBUS on the board: 10b: Port 3, Reserved 01b: Port 2, dedicated for TSI polling: 11b: Port 4, Reserved
18	**AsfClkSwitch**. Read-write. Reset: 0. 1=Change ASF master clock from RTC (32 kHz) to the clock defined in AsfClkSel.
17	**AsfClkStretchEn**. Read-write. Reset: 1. 1=Enable clock stretching support.
16	**AsfSmMasterEn**. Read-write. Reset: 0. 1=Enable ASF SMBus master function.
15:8	**SmbusAsfIoBase**. Read-write. Reset: 0Bh. Specifiies SMBus and ASF IO base address. • SMBus IO base = {Smbus0AsfIoBase[7:0], 0x00}. • ASF IO base = {Smbus0AsfIoBase[7:0], 0x20}. By default SMBus IO base is B00h and ASF IO base is B20h.
7	**WatchdogTmrEn**. Read-write. Reset: 0. 1=Enable Watchdog Timer memory (FEB0_0000h ~ FEB0_000Fh) decoding.
6	**HpetEn**. Read-write. Reset: 0. 1=Enable HPET memory (FED0_0000h ~ FED0_01FFh) decoding.
5	**IoApicEn**. Read-write. Reset: 0. 1=Enable IOAPIC memory (FEC0_0000h ~ FEC0_007Fh) decoding.
4	**SmbusAsfIoEn**. Read-write. Reset: 0. 1=Enable SMBus and ASF IO decoding. SMBUS and ASF IO range are defined in Smbus0AsfIoBase.
3	**DmaPort80**. Read-write. Reset: 0. 1=Pass IO port 80h, 81h, 82h, 83h to legacy DMA IO range.

2	**LegacyDmaIoEn**. Read-write. Reset: 0. 1=Enable legacy DMA IO range.
1	**Cf9IoEn**. Read-write. Reset: 0. 1=Enable CF9h IO port decoding.
0	**LegacyIoEn**. Read-write. Reset: 0. 1=Enable the following IO decoding: • 20h, 21h, A0h, A1h (PIC); • 40h, 41h, 42h, 43h, 61h (8254 timer); • 70h, 71h, 72h, 73h (RTC); • 92h.

PMx04 IsaControl

Bits	Description
31:17	Reserved.
16	**ABClkGateEn**. Read-write. Reset: 0. BIOS: See 2.17.8.2 [Global A-Link / B-Link Clock Gating]. Master switch for ALink and Blink clock gating. 0=Disabled. 1=Enabled.
15:6	Reserved.
5	**ReadShadow**. Read-write. Reset: 0. 1=Allow to read PIC ICWX, OCWX register through IO port 21h and A1h.
4	**LTEnable**. Read-write. Reset: 0. Specifies the legacy DMA transfer size.
3:2	Reserved.

1	**MmioEn**. Read-write. Reset: 0. BIOS: 1. 1=Enable Acpi MMIO range (FED8_0000h-FED8_1FFFh). The space is allocated as followed: 00FFh-0000h SMBus PCI configuration registers, see D14F0x00. 01FFh-0100h Reserved. 02FFh-0200h SMI, see SMIx00. 03FFh-0300h PMIO, see PMx00. 04FFh-0400h PMIO2, see PM2x00. 05FFh-0500h BIOS RAM. 06FFh-0600h CMOS RAM, see IO073_x00. 07FFh-0700h CMOS. 08FFh-0800h ACPI, see 3.26.14 [Standard ACPI Registers]. 09FFh-0900h ASF registers, see ASFx00. 0AFFh-0A00h SMBus registers, see SMBUSx00. 0BFFh-0B00h Watchdog registers, see WDTx00. 0CFFh-0C00h HPET, see HPETx000. 0DFFh-0D00h IOMux, see 3.26.11.2 [IOMux Registers]. 0EFFh-0E00h Miscellaneous registers, see MISCx00. 10FFh-1000h Reserved. 13FFh-1100h Reserved. 14FFh-1400h DP-VGA. 15FFh-1500h GPIO Bank 0. 16FFh-1600h GPIO Bank 1. 17FFh-1700h GPIO Bank 2. 1BFFh-1800h Reserved. 1CFFh-1C00h USB3 PHY, see XHCI_PMx00. 1DFFh-1D00h Wake Device (AC DC timer), see AcDcTimerx00. 1EFFh-1E00h Reserved. 1FFFh-1F00h Reserved.
0	**BiosRamEn**. Read-write. Reset: 0. 1=Enable BIOS RAM whose base is FED1_0000h (256 bytes).

PMx08 PciControl

Bits	Description
31:26	Reserved.
25	**ForceSlpStateRetry**. Read-write. Reset: 0. 1=Send out SMI message before the completion response of IO writes to AcpiPm1CntBlkx00[SlpTyp]. This is to be used in conjunction with SMI trapping on writes to AcpiPm1CntBlkx00[SlpTyp].
24	**ForceStpClkRetry**. Read-write. Reset: 1. 1=Send out STPCLK message before the completion response to the following 3 types of requests: • IO writes to AcpiPm1CntBlkx00[SlpTyp]. • IO write LDT_STP command. • C1e cycle. Normally it is required to send out STPCLK before completion of the cycles listed above, except for the case of SMI trapping. In the case of SMI trapping, this bit should be programmed as 0.
23	**AbStallEn**. Read-write. Reset: 0. 1=Allow the legacy DMA engine to hold the internal bus before completing legacy DMA on the LPC bus. This is only needed for certain old LPC devices.
22:21	Reserved.

20	**ShutDownOption**. Read-write. Reset: 0. 0=Issue Init message upstream when receiving shutdown message. 1=Generate PCI reset when receiving shutdown message.
19	**MasterNoWait**. Read-write. Reset: 0. 1=ACPI PCI Master doesn't wait for Slave idle when it wants to request bus. 0=PCI Master will wait for Slave idle.
18	Reserved.
17	**ChangeDma**. Read-write. Reset: 0.
16	**GateDma**. Read-write. Reset: 0.
15	**ClockSlowMask**. Read-write. Reset: 0.
14:12	**ExtIntrTime**. Read-write. Reset: 0. Specifies the extended interrupt time in 2 microsecond intervals. This is used for preventing APU from re-entering C-state right away when it just breaks out from a C-state.
11	**DlyEn**. Read-write. Reset: 0.
10	**IgnrUsbSmiReq**. Read-write. Reset: 1.
9	**BlockAcpiS5IntrSt**. Read-write. Reset: 0.
8	**PicApicArbiter**. Read-write. Reset: 1. 1=Enable arbitration between PIC request and IOAPIC request.
7	**ForceSmafMatch**. Read-write. Reset: 0. 1=Enable STPGNT message matching to the expected SMAF.
6	**MtsAuto**. Read-write. Reset: 0. 1=Encode PIC interrupt request as legacy PIC ExtInt message type and PIC NMI request as legacy PIC NMI message type if ((IOAPIC is enabled) && (MtsSet == 0)).
5	**MtsSet**. Read-write. Reset: 1. 1=Encode PIC interrupt request as legacy PIC ExtInt message type and NMI request as legacy NMI message type. 0=Encode PIC interrupt request as ExtInt message type and NMI request as NMI message type.
4	**MsgIntrEnable**. Read-write. Reset: 0. 1=Deliver legacy PIC interrupt as message type.
3	**MaskMsgBmStsEn**. Read-write. Reset: 0. 1=Enable A20#, IGNNE#, INIT#, NMI, SMI# message delivery.
2	**FakeDmaEn**. Read-write. Reset: 0.
1	**UndoWrdtDone**. Read-write. Reset: 0.
0	**UndoDmaChange**. Read-write. Reset: 0.

PMx0C StpClkSmaf

Bits	Description
31	Reserved.
30:28	**TtSmaf**. Read-write. Reset: 5h. Specifies the system management action field for thermal throttling STPCLK message.
27	Reserved.
26:24	**NsSmaf**. Read-write. Reset: 5h. Specifies the system management action field for normal throttling STPCLK message.
23	Reserved.
22:20	**S3Smaf**. Read-write. Reset: 4h. Specifies the system management action field for S3 STPCLK message.

19	Reserved.
18:16	**S1Smaf**. Read-write. Reset: 3h. Specifies the system management action field for S1 STPCLK message.
15	Reserved.
14:12	**VfSmaf**. Read-write. Reset: 2h. Specifies the system management action field for VFID STPCLK message.
11	Reserved.
10:8	**C3Smaf**. Read-write. Reset: 1. Specifies the system management action field for C3 STPCLK message.
7	Reserved.
6:4	**C2Smaf**. Read-write. Reset: 0. Specifies the system management action field for C2 STPCLK message.
3	Reserved.
2:0	**S4S5Smaf**. Read-write. Reset: 6h. Specifies the system management action field for S4/5 STPCLK message.

PMx10 Power Reset Config

Bits	Description
15:6	Reserved.
5	**RtcClkChkEnS5Reg**. Read-write. Reset: 1. 0=RtcClkChk function is disabled. 1=RtcClkChk function depends on the strap pin. If the strapped value is 1, RtcClkChk function is enabled. The strap name is RtcClkChkEn_strap.
4	**SlpS3WaitRstAsrt**. Read-write. Reset: 1. Debug purpose.
3:2	**RstBlkSel[1:0]**. Read-write. Reset: 01b. Selects the new or old logic for reset signals. Old logic uses fch_pwr_detect. New logic uses fch_PwrRst. Bits Definition 00b Always use old logic for reset signals. 01b Use new logic for reset signals during S0I3 entry/exit, otherwise, use old logic. 10b Always use new logic for reset signals. 11b Always use new logic for reset signals.
1	**ToggleAllPwrGoodOnCf9**. Read-write. Reset: 0. 1=De-assert and then assert all PwrGood signals (for PG1, PG1a, PG2 and XHC) during CF9 reset. 0=During CF9 reset, PG1_PwrGood stay at high; PG1a_PwrGood, PG2_PwrGood and Xhc_PwrGood behavior depend on TogglePG1aPG2PGXHCOnCf9 register bit during CF9 reset.
0	**TogglePG1aPG2PGXhcOnCf9**. Read-write. Reset: 1. 1=De-assert and then assert PG1a_PwrGood, PG2_PwrGood and Xhc_PwrGood during CF9 reset. 0=PG1a_PwrGood, PG2_PwrGood and Xhc_PwrGood stay at high during CF9 reset.

PMx40 eSPIIntrCtrl

Bits	Description
31:24	Reserved.
23:0	**eSPIDevIntrMask**. Read-write. Reset: 0. Set these bits to mask of eSPI Device IRQ23~0. 1=Mask off the interrupt. 0=Enable eSPI IRQ23~0.

PMx44 BootTimerEn

Bits	Description
31	**BootTmrDisable**. Read-write. Cold reset: 0. BIOS: 1. 0=Boot timer starts running. 1=Boot timer is stopped; it will not trigger a system reset or de-assertion on the NbPwrGood. Software should set the bit to 1 after every reset or S3/S4/S5 resume before the timer expires in 1.17 seconds.
30	**FailBootRstSts**. Read; Write-1-to-clear. Cold reset: 0. 0=Boot timer has not been fired. 1=Boot timer has been fired.
29	**ExpireBootTmr**. Read-write. Cold reset: 0. 1=Force boot timer to expire; then NbPwrGood can be asserted.
28	**BootTmrStopOnGAlink**. Read-write. Cold reset: 1. BIOS: 0. 0=Enable the boot timer to ensure a good boot. 1=Boot timer stops when FCH observes a good boot after PCI reset or S3/S4/S5 resume.
27	**BootTmrFuncEn**. Read-write. Cold reset: 1. BIOS: 1. 0=Disable boot timer function; avoid system restarts when performing BIOS debug. 1=Enable boot timer function; the boot timer starts to count down and toggles NbPwrGood after 1.17 seconds if software has not set ExpireBootTmr to 1 after PCI reset or resuming from S3/S4/S5.
26:25	Reserved.
24:0	**FailBootTimer**. Read-write. Cold reset: 0. Specifies the counter of APU boot timer (14.318 MHz), which starts counting when both of the following conditions are met: • BootTmrDisable == 1. • PCI reset is not asserted.

PMx48 PGPwrEnDly

Bits	Description
31:24	**PG1PwrDownDlyTmr**. Read-write. Reset: 51h. PG1 Power Down delay timer.
23:16	**XhcPwrEnDlyTmr**. Read-write. Reset: 41h. XhcPwrEn delay timer.
15:8	**PG2PwrEnDlyTmr**. Read-write. Reset: 31h. PG1aPwrEn delay timer.
7:0	**PG1aPwrEnDlyTmr**. Read-write. Reset: 21h. PG1aPwrEn delay timer.

PMx54 SerialIrqConfig

Bits	Description
15:11	Reserved.

10	**UndoSerIrqChange**. Read-write. Reset: 0.
9	**LegacyDis**. Read-write. Reset: 0.
8	**SirqCntrl**. Read-write. Reset: 0.
7	**SerialIrqEnable**. Read-write. Cold reset: 0. 1=Enable the serial IRQ function.
6	**SerIrqMode**. Read-write. Cold reset: 0. 0=Continuous mode. 1=Active (quiet) mode.
5:2	**NumSerIrqBits**. Read-write. Cold reset: 0. Specifies the total number of serial IRQs. Bits Definition 0h 17 serial IRQs (15 IRQ#, SMI#, IOCHK#) 1h 18 serial IRQs (15 IRQ#, SMI#, IOCHK#, INTA#) Fh-2h <NumSerIrqBits+17> The serial IRQ can support 15 IRQ#, SMI#, IOCHK#, INTA#, INTB#, INTC#, and INTD#. When serial SMI# is used,
1:0	**NumStartBits**. Read-write. Cold reset: 0. This field specifies the number of clocks in the start frame. Start Frame Width = 4 + 2 * NumStartBits.

PMx56 RTC Control

Bits	Description
15	Reserved.
14	**ExtraRtcCmosEn**. Read-write. Cold reset: 0. 1=Software can access the extra 16 bytes of RTC CMOS RAM through IO072 and IO073; The extra RAM space are located at index 03h:00h, 43h:40h, 83h:80h and C3h:C0h. 0=Access to the extra RTC CMOS RAM space is disabled.
13	**AltCmosMapEn**. Read-write. Cold reset: 0. 1=When accessing the RTC CMOS RAM through IO070 and IO071, Bank 1 of CMOS RAM is changed: Index 0Dh:00h still returns the time and alarm settings; Index 7Fh:0Eh returns the absolute offset FFh:8Eh which map to the extended RAM space.
12	**CenturyEn**. Read-write. Cold reset: 1. 1=Enable RTC Century support.
11	**MaskRtcClkOut**. Read-write. Cold reset: 0. 0=Disable RtcClk output.
10	**RtcClkDrive**. Read-write. Cold reset: 1. 0=HIGHDRIVE is tied low for RtcClkOut pad. 1=HIGH-DRIVE is tied high for RtcClkOut pad.
9:5	Reserved.
4	**RtcProtectC0_CF**. Write-once. Cold reset: 0. 1=RTC RAM index CFh:C0h are locked from read/write.
3	**RtcProtectD0_DF**. Write-once. Cold reset: 0. 1=RTC RAM index DFh:D0h are locked from read/write.
2	**RtcProtectE0_EF**. Write-once. Cold reset: 0. 1=RTC RAM index EFh:E0h are locked from read/write.
1	**RtcProtectF0_FF**. Write-once. Cold reset: 0. 1=RTC RAM index FFh:F0h are locked from read/write.
0	**RtcProtect38_3F**. Write-once. Cold reset: 0. 1=RTC RAM index 3Fh:38h are locked from read/write.

PMx58 VRT_T1

Bits	Description
7:0	**VRT_T1**. Read-write. Cold reset: 1. This field specifies the time of VRT_Enable being high for RTC battery monitor circuit in milliseconds. To conserve power, the RTC battery is sampled periodically for checking its state of health. VRT_T1 and PMx59[VRT_T2] make up the interval of the checking. When VRT_Enable is high, the battery is being sampled. When VRT_Enable is low, the battery is not being sampled.

PMx59 VRT_T2

Bits	Description
7:0	**VRT_T2**. Read-write. Cold reset: FFh. This field specifies the time of VRT_Enable being low for the RTC battery monitor circuit in 4 ms increments. See PMx58 [VRT_T1] for detailed description.

PMx5B RTC Shadow

Bits	Description
7:4	Reserved.
3:0	**PwrFailShadow**. Read-write. Cold reset: 0. BIOS: 0100b.

PMx5C LLBCntrl

Bits	Description
7:3	Reserved.
2	**AllowWakeS3En**. Read-write. Cold reset: 0. 1=Allow LLB# as wake event in S3. Note: LLB (Low low battery) event should trigger a SCI if system is in S0. It should block wake-up if the system is in S-States. To meet that required behavior, software should use the following settings: • PMxC8[LlbEn] = 1. • BlockWakeEn = 0. • UseAsWakeEn = 1. • AllowWakeS3En = 0 or 1 (don't care).
1	**UseAsWakeEn**. Read-write. Cold reset: 0. 1=Treat LLB# as wake event.
0	**BlockWakeEn**. Read-write. Cold reset: 0. 1=Block wake event if LLB# is asserted; if (UseAsWakeEn == 1) && (AllowWakeS3En == 1), LLB# and other wake events can wake the system up from S3.

PMx5E RTC ExtIndex

Bits	Description
7:0	**Index**. Read-write. Cold reset: X. Specifies the offset of the RTC Extended Register to be read/written from PMx5F.

PMx5F RTC ExtData

Bits	Description
7:0	**Data**. Read-write. Cold reset: X. Specifies the read data or write data of the RTC Extended Register.

PMx5F_x00 RTCEXT DltSavEnable

Bits	Description
7:1	Reserved.
0	**DltSavEnable**. Read-write. Reset: 0. 1=Enable RTC daylight saving feature.

PMx5F_x01 RTCEXT SprFwdCtrl

Bits	Description
7	Reserved.
6	**SprFwdWeek**. Read-write. Reset: 0. This specifies which Sunday morning to do the “Spring forward”. Spring forward is usually at the last Sunday of March in Europe. 0=The 1st Sunday of the month. 1=The last Sunday of the month.
5:0	**SprFwdHour**. Read-write. Reset: 0. This Binary-Coded Decimal (BCD) value specifies which hour (24-hour mode) to do the “Spring forward”. Spring forward is usually 2 am in United States and 1 am in Europe. 0=2 am. 01h=1 am. 02h=2 am.

PMx5F_x02 RTCEXT SprFwdMonth

Bits	Description
7:5	Reserved.
4:0	**SprFwdMonth**. Read-write. Reset: 0. This Binary-Coded Decimal (BCD) value determines which month to do the “Spring forward”. Spring forward is usually in March in United States and Europe. 0=April. 03h=March. 04h=April.

PMx5F_x03 RTCEXT FallBackCtrl

Bits	Description
7	Reserved.
6	**FallBackWeek**. Read-write. Reset: 0. This value specifies which Sunday morning to do the “Fall back”. Fall back is usually at the last Sunday of October in Europe and the first Sunday of November in the United States. 0=The last week of the month. 1=The first week of the month.
5:0	**FallBackHour**. Read-write. Reset: 0. This Binary-Coded Decimal (BCD) value specifies which hour (24-hour mode) to do the “Fall back”. Fall back is usually 2 am in United States and 1 am in Europe. 0=2 am. 01h=1 am. 02h=2 am.

PMx5F_x04 RTCEXT FallBackMonth

Bits	Description
7:5	Reserved.
4:0	**FallBackMonth**. Read-write. Reset: 0. This Binary-Coded Decimal (BCD) value specifies which month to “Fall back”. Fall back is usually the first week of November in The United States and last week of October in Europe. 0=October. 10h=October. 11h=November.

PMx5F_x10 RTCEXT WeekTimerControl

The 16-bit Week Timer is a battery-powered down counter timer that supports 1 ms, 1 second, and 1 minute resolution and auto reloads when the timer reaches 0. The WEEK_ALRM interrupt is asserted when the timer reaches 0 and stays asserted until the timer is disabled.

Bits	Description
7:3	Reserved.
2:1	**Resolution**. Read-write. Reset: 0. This field specifies the resolution of the Week Timer counter. Before programing this bit, software should program Enable to 0 to disable the Week Timer. Bits Definition Bits Definition 00b 1 minute 10b 1 ms 01b 1 second 11b Reserved
0	**Enable**. Read-write. Reset: 0. 0=Disable Week Timer. 1=Enable Week Timer.

PMx5F_x11 RTCEXT WeekTimerReloadLow

Bits	Description
7:0	**WeekTimerReloadLow**. Read-write. Reset: 0. This filed is used to program the lower 8 bits of the 16-bit WeekTimerReload register. Writing the WeekTimerReloadLow register causes the 16-bit WeekTimerReload to be written into the Week Timer. Software should program PMx5F_x10[Enable] = 0 before writing to this register.

PMx5F_x12 RTCEXT WeekTimerReloadHigh

Bits	Description
7:0	**WeekTimerReloadHigh**. Read-write. Reset: 0. This field is used to program the upper 8 bits of the 16-bit WeekTimerReload register. This field should be programmed before programming PMx5F_x11 [RTCEXT WeekTimerReloadLow]. Software should program PMx5F_x10[Enable] = 0 before writing to this register.

PMx5F_x13 RTCEXT WeekTimerDataLow

Bits	Description
7:0	**WeekTimerDataLow**. Read-only. Reset: 0. This field is used to read the current state of the 16-bit Week Timer. Reading from the WeekTimerDataLow register returns the lower 8 bits of the 16-bit Week Timer and causes the upper 8 bits to be latched into PMx5F_x14 [RTCEXT WeekTimerData-High].

PMx5F_x14 RTCEXT WeekTimerDataHigh

Bits	Description
7:0	**WeekTimerDataHigh**. Read-only. Reset: 0. This field is used to read the current state of the 16-bit Week Timer. Reading from the WeekTimerDataHigh register returns the upper 8 bits of the 16-bit Week Timer latched by a previous read from PMx5F_x11 [RTCEXT WeekTimerReloadLow].

PMx60 AcpiPm1EvtBlk

Bits	Description
15:2	**AcpiPm1EvtBlk**. Read-write. Cold reset: 0. Specifies the 16-bit IO range base address[15:2] of the ACPI power management event block defined in 3.26.14.1 [AcpiPmEvtBlk].
1:0	Reserved.

PMx62 AcpiPm1CntBlk

Bits	Description
15:1	**AcpiPm1CntBlk**. Read-write. Cold reset: 0200h. Specifies the 16-bit IO base address[15:1] of the ACPI power management control block defined in 3.26.14.2 [AcpiPm1CntBlk].
0	Reserved.

PMx64 AcpiPmTmrBlk

Bits	Description
15:1	**AcpiPmTmrBlk**. Read-write. Cold reset: 0. Specifies the 16-bit IO base address[15:1] of the ACPI power management timer block defined in 3.26.14.4 [AcpiPmTmrBlk].
0	Reserved.

PMx66 CpuCntBlk

Bits	Description
15:3	**CpuControl**. Read-write. Cold reset: 0. Specifies the 16-bit IO base address[15:3] of the ACPI power management CPU control block defined in 3.26.14.5 [CpuCntBlk].
2:0	Reserved.

PMx68 AcpiGpe0Blk

Bits	Description
15:2	**AcpiGpe0Blk**. Read-write. Cold reset: 0. Specifies the 16-bit IO base address[15:2] of the ACPI power management general purpose event block defined in 3.26.14.6 [AcpiGpe0Blk].
1:0	Reserved.

PMx6A AcpiSmiCmd

Bits	Description
15:0	**AcpiSmiCmd**. Read-write. ColdReset: 00B0h. These bits define the 16-bit IO base address[15:0] of the ACPI SMI command block defined in 3.26.14.7 [SmiCmdBlk]. The address is required to be WORD-aligned (Addr[0] = 0).

PMx6E AcpiPm2CntBlk

Bits	Description
15:0	**AcpiPm2CntBlk**. Read-write. Cold reset: 0. These bits define the 16-bit IO base address[15:0] of the ACPI power management additional control block defined in 3.26.14.3 [AcpiPm2CntBlk].

PMx74 AcpiConfig

Bits	Description
31:30	Reserved.
29	**RtcWakeAlarm**. Read-write. Cold Reset: 0. 0=Use IO073_x0C[IRQF] as one of the system wake-up event if IRQF is enabled. 1=Use IO073_x0C[AF] as one of the system wake-up event if AF is enabled.
28	**PcieGeventMap**. Read-write. Cold reset: 1. 1=Route PME message from NB to GEvent 24, hot plug message from APU to GEvent 7.
27	**WakePinAsGevent**. Read-write. Cold reset: 0. 1=Treat Wake# pin as GEvent input.
26	Reserved.
25	**PcieWakMask**. Read-write. Cold reset: 0. 1=Disable the PCIE_WAK_STS and PCIE_WAK_DIS function defined in AcpiPmEvtBlkx00[PciExpWakeStatus] and AcpiPmEvtBlkx02[PciExpWakeDis].
24	**PcieNative**. Read-write. Cold reset: 0. 1=Block PCIe® GPP PME message and hot plug message from generating SCI.
23:10	Reserved.
9	**AcpiReducedHwEn**. Read-write. Reset: 0. 0=ACPI fixed register interface is enabled. 1=The decoding of ACPI fixed registers and SCI are disabled; In addition wake function from AcpiPmEvtBlk is disabled as well.
8	**PwnBtnEn**. Read-write. Reset: 1. 1=Enable power button support in AcpiPmEvtBlk block.
7	**BiosRls**. RAZ; Write-1-only. Cold reset: 0. Writing 1 to this bit generates SMI, NMI or IRQ13 depending on SMIxB0[Smicontrol73].

6	**MaskArbDis**. Read-write. Cold reset: 1.
5	Reserved.
4	**TmrEnEn**. Read-write. Cold reset: 0. 1=Enable the TMR_EN function defined in AcpiPmEvtBlkx02[TmrEn].
3	Reserved.
2	**RtcEnEn**. Read-write. Cold reset: 0. 1=Enable the RTC_EN function defined in AcpiPmEvtBlkx02[RtcEn].
1	**GblEnEn**. Read-write. Cold reset: 0. 1=Enable the GBL_EN function defined in AcpiPmEvtBlkx02[GblEn].
0	**DecEnAcpi**. Read-write. Cold reset: 0. 1=Enable decoding of the standard ACPI registers.

PMx78 WakeIoAddr

Bits	Description
15:0	**WakeIoBaseAddress**. Read-write. Reset: FFFFh.

PMx7A HaltCountEn

Bits	Description
15	**CountHaltMsgEn**. Read-write. Reset: 0. 1=FCH keeps track of the APU state by counting HALT entering and exit messages; When the number of net HALT enter messages matches with NumOfHalt, FCH initiates C1e.
14:4	Reserved.
3:0	**NumOfHalt**. Read-write. Reset: 0. Defines the number of HALT messages to track before FCH should initiate C1e sequence.

PMx7C C1eWrPortAdr

Bits	Description
15:0	**C1eWrPortAdr**. Read-write. Reset: FFFFh.

PMx7E CStateEn

Bits	Description
15	**MaskIntrEn**. Read-write. Reset: 1. 1=APIC interrupts are deferred until the first ACPI access after the system resumes from S state; In addition, A20M, IGNNE, INTR, NMI, INIT messages are deferred the same way, but SMI is not be deferred. This is mainly to guard against unexpected interrupts being sent to the OS during S1 resume.
14	**MaskCStateSys**. Read-write. Reset: 0. 1=Skip C-state if there is a pending interrupt such as SMI, NMI, and INIT request.
13:8	Reserved.
7	**CMsgMaskEn**. Read-write. Reset: 1.

6	**CPopUpEn**. Read-write. Reset: 1. 1=Enable pop up capability, which means APU goes to C2 if there is a traffic and comes back to C3 after being idle for a while. This bit should be set to 0 when AltVid is not enabled.
5	**C1eToC3En**. Read-write. Reset: 1. 1=Put APU into C3 state in C1e state.
4	**C1eToC2En**. Read-write. Reset: 0. 1=Put APU into C2 state in C1e state.
3	**C2EnhanceEn**. Read-write. Reset: 0. 1=Enable C2 enhancement.
2	**C2ToC3Enable**. Read-write. Reset: 0. 1=Put APU into C3 even in the case of IO reads to CpuCntBlkx04 [PLvl2].
1:0	**C1eWrPortSel**. Read-write. Reset: 00b.

PMx80 Break Event

Bits	Description
31	Reserved.
30:24	**AutoStutterLimit**. Read-write. Cold reset: 0. This specifies the limit for the AutoStutterTimer. Time unit is defined by [AutoStutterTimeSel]. See [AutoStutterTimerEn] for detailed description on AutoStutterTimer.
23:22	Reserved.
21	**UsbPeriodicalSetBmSts**. Read-write. Cold reset: 1. 1=Treat USB isochronous traffic as source of DMA traffic.
20:19	Reserved.
18	**Usb20SetBmSts**. Read-write. Cold reset: 1. 1=Treat USB 2.0 traffic as source of DMA traffic.
17:14	Reserved.
13	**ServerCEn**. Read-write. Cold reset: 0. 1=FCH monitors the HALT messages coming from APU(s). When all APUs are in HALT state, FCH automatically transitions to C3/C1e state. If any APU exits from HALT state, FCH exits out from C3/C1e state as well. This is used in conjunction with PMx7A.
12	**AutoStutterTimeSel**. Read-write. Cold reset: 0. This bit selects the time increment used by the AutoStutterTimer. 0=2 us increment. 1=1 ms increment.
11	**AutoStutterTimerEn**. Read-write. Cold reset: 0. 1=Enable an AutoStutterTimer that automatically counts whenever LDTSTOP is asserted during C3/C1e state. When the timer reaches the threshold defined by AutoStutterLimit, it stutters the C-state machine. This feature is specifically designed to periodically reconnect the HyperTransport™ link in the case of a long idle time for multi-CPUs.
10	**MergeBMReqEn**. Read-write. Cold reset: 0. 1=The logic merges BMREQ# and AllowLdtStop together internally.
9	**BusReqHoldEn**. Read-write. Cold reset: 0. 1=Extend BMREQ# until LDTSTP# is asserted.
8	**BmReqPopUpEn**. Read-write. Cold reset: 0. 1=Allow PopUp if BMREQ# is toggled when PopUp function is enabled.
7	**BmReqEn**. Read-write. Cold reset: 0. 1=Treat BMREQ# as one source of Break Event.
6	Reserved.
5	**EffMaskEn**. Read-write. Cold Reset: 0.
4	**EnableBreak**. Read-write. Cold reset: 0. 1=Skip the C-state transition if there is break event when entering C-state.

3	**MaskNbBmStsSet**. Read-write. Reset: 0.
2	**AutoClrBmSts**. Read-write. Cold reset: 0. It is used for PopUp and C1e function. 1=Automatically clear AcpiPmEvtBlkx00[BmStatus] before entering C1e state.
1	**AutoBmRld**. Read-write. Cold reset: 0. 1=Generate an internal AcpiPmEvtBlkx00[BmStatus] enable bit (that is similar to AcpiPm1CntBlkx00[BmRld]) upon entry to C1e. Depending on the configuration of other bits, bus master activity or IDLE_EXIT# pin could cause FCH to break out from C1e.
0	**BmStsRdMask**. Read-write. Cold reset: 0. 0=AcpiPmEvtBlkx00[BmStatus] is set to1 if there is any DMA traffic in FCH. 1=Make AcpiPmEvtBlkx00[BmStatus] always return 0 except for USB 1.1 traffic.

PMx88 CStateControl

Bits	Description
31:14	Reserved.
13	**LdtStpCmd**. Read-write. Reset: 0. Programming it to 1 from 0 forces FCH to toggle LDTSTP#.
12	**StutterMode**. Read-write. Reset: 0. 1=Enable stutter mode.
11	**NoChkAgpBusyEnB**. Read-write. Reset: 0.
10:9	Reserved.
8	**AgpStpEn**. Read-write. Reset: 0.
7:6	Reserved.
5	**SlpEn**. Read-write. Reset: 0. 1=Enable LDTSTOP# as an output.
4	**CcEn**. Read-write. Reset: 0. 1=Enable C-state. This bit must be set in order to exercise C-state.
3	Reserved.
2	**DlySlpEn**. Read-write. Reset: 0. 1=Delay recognition of STPGNT# until there is no pending read in AB.
1	Reserved.
0	**WaitStpGntEnB**. Read-write. Reset: 0. 1=Wait for STPGNT# in ACPI S-state.

PMx8E PopUpEndTime

Bits	Description
7:0	**PopUpEndTime**. Read-write. Cold reset: 80h.

PMx94 CStateTiming0

Bits	Description
31:26	Reserved.
25:24	**LdtEndTime**. Read-write. Cold reset: 0. Specifies the delay from deassertion of LDTSTP# till the deassertion of STPCLK. Bits Definition Bits Definition 00b 0 us 10b 32 us 01b 1 us 11b 64 us

23:16	**LdtStartTime**. Read-write. Cold reset: 10h. Specifies the LDTSTP# deassertion time in 1 us increments in C-state.
15:8	**SLdtStartTime**. Read-write. Cold reset: 0. Specifies the delay between LPC_PD# assertion and LDTSTP# assertion when the system enters ACPI S states. The time is in 1 us increments, with 1 us uncertainty.
7:0	**StutterTime**. Read-write. Cold reset: 1. Specifies the LDTSTP# duration in 1 us increments. This is basically the minimum LDTSTP# assertion time.

PMx98 CStateTiming1

Bits	Description
31	Reserved.
30:28	**VidFidTime**. Read-write. Cold reset: 0. Specifies the VID/FID LDTSTP# duration. Bits Definition Bits Definition 000b 0 us 100b 16 us 001b 2 us 101b 32 us 010b 4 us 110b 64 us 011b 8 us 111b 128 us
27:26	Reserved.
25:24	**AgpEndTime**. Read-write. Cold reset: 01b.
23:16	Reserved.
15:8	**FirstLdtStartTime**. Read-write. Cold reset: 10h. Specifies the very first LDTSTP# assertion delay in 1 us increments from reception of STPGNT.
7:0	**StpClkDlyTime**. Read-write. Cold reset: 0. Specifies the additional STPCLK# deassertion delay in number of oscillator clocks for S1 resume.

PMx9C C2Count

Bits	Description
7:0	**C2Count**. Read-only; Updated-by-hardware. Reset: 0. Specifies the amount of time the APU spends in STPGNT state (but LDTSTOP# is not asserted) during C1e. Each increment is approximately 0.39% (1/256). This field is updated every second.

PMx9D C3Count

Bits	Description
7:0	**C3Count**. Read-only; Updated-by-hardware. Reset: 0. Specifies the amount of time LDTSTOP# is asserted during C1e. Each increment is approximately 0.39% (1/256). This field is updated every second.

PMxA0 MessageCState

Bits	Description
31	**CheckCoreIdDis**. Read-write. Reset: 0. 1=Ignore core ID check if (MultiCoreEn == 1). 0=Enable core ID check if (MultiCoreEn == 1).
30:24	**ExtendValue**. Read-write. Reset: 0. Specifies the timer value to be used with ExtendEnable. The value is the number of 66 MHz clocks.
23	**MultiCoreEn**. Read-write. Reset: 0. 1=C-state control logic inside FCH assumes multi-core configuration; PMx7A[NumOfHalt] should be programmed accordingly; FCH keeps track of APU C-state by monitoring each core's C-state message instead of package state.
22	**FchExitCMsgDis**. Read-write. Reset: 0. 1=ExitC Message is blocked.
21	**FchAllowCMsgDis**. Read-write. Reset: 0. 1=AllowC Message is blocked.
20:16	**TmrSelOverride**. Read-write. Reset: 0. These bits are used with TimerTickChgMsgEn. In case that FCH auto-timer detection logic doesn't function properly, these bits can be used to override the logic and force the logic to monitor the specific timer. Bit — Definition [4] — 1=Force the logic to monitor HPET timer 2 if HPET is selected [3] — 1=Force the logic to monitor HPET timer 1 if HPET is selected [2] — 1=Force the logic to monitor HPET timer 0 if HPET is selected [1] — 1=Use RTC [0] — 1=Use HPET
15:14	**UsbEhciModeFch[1:0]**. Read-write. Reset: 0. These bits are used with FCH CPU C-state. Bit — Definition [1] — 1=FCH causes CPU to break out from C-state when USB EHCI has pending traffic. [0] — 1=FCH stutters the CPU C-state whenever USB EHCI has pending traffic.
13:12	**UsbOhciModeFch[1:0]**. Read-write. Reset: 0. These bits are used with FCH CPU C-state. Bit — Definition [1] — 1=FCH causes CPU to break out from C-state when USB OHCI has pending traffic. [0] — 1=FCH stutters the CPU C-state whenever USB OHCI has pending traffic.
11	**ExtendEnable**. Read-write. Reset: 0. 1=Enable FCH to start a timer whenever the C-state is exited, in order to prevent CPU from going back to C-state before the timer expires. See ExtendValue for the timer.
10	**ClkIntrTagEn**. Read-write. Reset: 0. 1=Enable FCH to mark the periodic timer interrupt.
9	**PopUpByEc**. Read-write. Reset: 0. A static bit that can be written by Integrated Micro-Controller (IMC) to cause FCH to pop up C-state message to CPU.
8	**WakeByEc**. Read-write. Reset: 0. A static bit that can be written by IMC to cause FCH to generate C-state exit message to CPU.
7	**SelfExitEnable**. Read-write. Reset: 0. 1=FCH exits to C0 state from non-C0 state when there is a break event, and exits to C1 state from non-C0 state when there is traffic.
6	**BatteryModeEn**. Read-write. Reset: 0. 1=A change in power mode (battery vs AC) causes FCH to tell CPU to exit from C-state.
5	Reserved.

4	**FchSerrEn**. Read-write. Reset: 0. 1=FCH C-state coordination logic causes CPU to exit from C-state when there is a system error within the FCH.
3	**FchPerrEn**. Read-write. Reset: 0. 1=FCH C-state coordination logic causes CPU to exit from C-state when there is a parity error within the FCH.
2	**FchCEnable**. Read-write. Reset: 1=Enable the FCH CPU C-state coordination logic. FCH CPU contains the actual C-state logic and FCH contains the coordination logic which sends handshake message to CPU to help it to decide which C-state to go into.
1	**TimerTickChgMsgEn**. Read-write. Reset: 0. 1=FCH sends a message to APU indicating the latest periodic timer interval. FCH automatically determines which timer (PIT, RTC, or HPET) is being used. See TmrSelOverride.
0	**BattModeChgMsgEn**. Read-write. Reset: 0. 1=FCH automatically sends a message to CPU indicating the power mode (AC vs battery); In addition, every time it is changed, the FCH generates a message to indicate the update.

PMxB0 DeferTimeTick

Bits	Description
31:11	Reserved.
10:8	**DeferTimerTickValue**. Read-write. Reset: 0. Bits Definition Bits Definition 000b No skipping 100b Skip 4 timer ticks 001b Skip 1 timer tick 101b Skip 5 timer ticks 010b Skip 2 timer ticks 110b Skip 6 timer ticks 011b Skip 3 timer ticks 111b Skip 7 timer ticks
7:2	Reserved.
1	**ForceTmrTickEn**. Read-write. Reset: 0. 1=If (DeferTimerTickEn == 1) && FCH has skipped a timer tick interrupt, FCH immediately generates the timer tick interrupt upon C-state exit.
0	**DeferTimerTickEn**. Read-write. Reset: 0. 1=FCH skips a number of timer tick interrupts based on the value defined in DeterTimeTickValue when CPU is in C state. When CPU is not in C state, FCH does not skip any timer tick interrupt.

PMxB4 Tpreset1b

Bits	Description
31:30	Reserved.
29:24	**Tpreset1b**. Read-write. Cold reset: 5h. This is the timing parameter used for S*->S0 state transition. It specifies the delay between CPU_STP# de-assertion and LPC_PD# de-assertion, in 8 us increment with 8 us uncertainty.
23:12	Reserved.
11:9	**FidVidOption**. Read-write. Cold reset: 0. Specifies the additional FID/VID exit delay. Bits Definition Bits Definition 000b 0 ns 100b 490 ns 001b 140 ns 101b 560 ns 010b 210 ns 110b 350 ns 011b 280 ns 111b 420 ns

8	**DelayLdtStp**. Read-write. Cold reset: 0. 1=Enable LDTSTP# assertion time.
7:1	Reserved.
0	**FidProtectEn**. Read-write. Cold reset: 0. 1=Skip C-state transition when FID/VID messages are received concurrently.

PMxB8 Tpreset2

Bits	Description
7:6	**ClkGateCntrl**. Read-write. Cold reset: 2h. These two bits control whether SMBUS module allows clock gating to the internal 66 MHz core clock. Bits — Definition 00b — Disable the clock gating function. 01b — Wait 16 clocks before allowing clock gating to the SMBUS module. 10b — Wait 64 clocks before allowing clock gating to the SMBUS module. 11b — Wait 256 clocks before allowing clock gating to the SMBUS module.
5:0	**Tpreset2**. Read-write. Cold reset: 8h. This is the timing parameter used for S*->S0 state transitions. It specifies the LDTSTP# deassertion delay in 8 us increment with 8 us uncertainty.

PMxB9 LpcMisc

Bits	Description
7:5	Reserved.
4	**AutoSizeDone_pmio**. Read-write. Cold reset: 0. AutoSizeDone sticky bit is used with save-restore. Before S0I3, software needs to read LPC config register D14F3xC8[AutoSizeStart] and write the value in this bit.
3	**ClkRunDisable**. Read-write. Cold reset: 0. Legacy DMA and serial IRQ logic reside in this module and they are running on the 33 MHz LPCCLK. 1=Disable this module's ability to support CLKRUN# function from PCIBridge; In other words, this module prevents PCIBridge from stopping the 33 MHz clock.
2:0	Reserved.

PMxBA S_StateControl

Bits	Description
15	**MaskPmeMsgEn**. Read-write. Cold reset: 0. 1=If (PmeMsgEn == 1), PmeAck messages are ignored and ACPI S-state logic solely uses the timeout mechanism to sequence through the S3 state.
14	**WakePinEnable**. Read-write. Cold reset: 0. 1=Enable wake up from WAKE# pin.
13:6	Reserved.
5	**SlpWaitTmrOutEn**. Read-write. Reset: 1. 0=Sleep sequence wait for USB3 and IMC completion. 1=Enable 15 ms time out when waiting for USB3 and IMC completion in sleep shutdown sequence.
4	**PmeMsgTrig**. Read-write. Reset: 0. IF (WRITE) THEN Software writes this bit to trigger a Pme-TurnOff sequence to NB. IF (READ) THEN Reading this bit returns the status of the PmeTurnOff sequence. 1=Not done. 0=Done.

3	**PmeMsgEn**. Read-write. Cold reset: 0. 1=Enable PmeTurnOff/PmeMsgAck handshake.
2:1	Reserved.
0	**LongSLPS3**. Read-write. Cold reset: 0. 1=Extend SLP_S3# assertion to 1 second minimum.

PMxBC ThrottlingControl

Bits	Description
15	**Therm2SecDelay**. Read-write. Cold reset: 0. 1=Enable 2 second delay for thermal clock throttle. This bit affects both hardware and software throttle.
14	**NoWaitStpGntEn**. Read-write. Cold reset: 0. 0=Wait for STPGNT after asserting STPCLK. 1=Do not wait for STPGNT after asserting STPCLK. This bit affects both hardware and software throttle.
13	**ThermThrotPeriod**. Read-write. Cold reset: 0. Specifies the clock throttle period for hardware thermal throttle. 0=30 us. 1=244 us.
12:8	Reserved.
7	**ThrottleControl[3]**. Read-write. Cold reset: 0. 1=Enable hardware thermal clock throttle.
6:4	**ThrottleControl[2:0]**. Read-write. Cold reset: 0. Specifies the throttle interval for STPCLK de-assertion in hardware thermal clock throttle: Bits Definition Bits Definition 000b 50% 100b 50% 001b 12.5% 101b 62.5% 010b 25% 110b 75% 011b 37.5% 111b 87.5%
3:2	Reserved.
1:0	**AcpiThrotPeriod**. Read-write. Cold reset: 0. Specifies the clock throttle period for software thermal throttle. See CpuCntBlkx00 [ClkValue] for software thermal throttle. Bits Definition Bits Definition 00b 15 us 10b 244 us 01b 30 us 11b Reserved

PMxBE ResetControl1

Table 282: BIOS Recommendations for KbRstEn

Case	Condition	Setting	Comments
1	If the KBRST# / GEVENT1# I/O pin is not connected to system keyboard reset or is configured as GEvent1 function.	0	This bit must be cleared by the platform system BIOS in this case.
2	All other situations other than that specified in Case 1.	Do not program.	Value should remain at power-up default setting.

Bits	Description
7	**RstToCpuPwrGdEn**. Read-write. Cold reset: 0. 1=FCH toggles CpuPwrGd on every reset.

6	**HwmResetOption**. Read-write. Cold reset: 1. 0=Hwm function defined at 3.26.13 [Power Management Block 2 (PM2) Registers] is reset by RsmRst. 1=Hwm function defined at 3.26.13 [Power Management Block 2 (PM2) Registers] is reset by PciRst.
5	**SlpTypeControl**. Read-write. Cold reset: 1. 1=Enable the function of AcpiPm1CntBlkx00[SlpTypeEn]. 0=AcpiPm1CntBlkx00[SlpTypeEn] bit has no effect.
4	**KbRstEn**. Read-write. Cold reset: 1. BIOS: See Table 282. 1=Enable KBRST# pin to trigger keyboard reset.
3:2	**CpuRstControl**. Read-write. Cold reset: 0. Bits Definition 00b CpuReset is deasserted after PciReset. 01b CpuReset is deasserted as PciReset. 10b CpuReset is deasserted before PciReset. 11b CpuReset is deasserted after PciReset.
1	**KbPciRstEn**. Read-write. Cold reset: 1. This bit must not be programmed by the BIOS. It should be left with the power up default value of 1. 1=Make PCI reset during keyboard reset, which can be triggered by KBRST# pin or IMC.
0	**SoftResetEn**. Read-write. Cold reset: 0. 1=Block any reset request until the system is not in C state.

PMxC0 S5/Reset Status

This register shows the source of previous reset.

Bits	Description
31:30	Reserved.
29	**EcWatchdogRst**. Read; Write-1-to-clear. Cold reset: 0.
28	**HangReset**. Read; Write-1-to-clear. Cold reset: 0.
27	**SyncFlood**. Read; Write-1-to-clear. Cold reset: 0.
26	**RemoteResetFromASF**. Read; Write-1-to-clear. Cold reset: 0.
25	**WatchdogIssueReset**. Read; Write-1-to-clear. Cold reset: 0.
24	**FailBootRst**. Read; Write-1-to-clear. Cold reset: 0.
23	**LtReset**. Read; Write-1-to-clear. Cold reset: 0.
22	**KbReset**. Read; Write-1-to-clear. Cold reset: 0.
21	**SleepReset**. Read; Write-1-to-clear. Cold reset: 0.
20	**DoFullReset**. Read; Write-1-to-clear. Cold reset: 0.
19	**DoReset**. Read; Write-1-to-clear. Cold reset: 0.
18	**DoInit**. Read; Write-1-to-clear. Cold reset: 0.
17	**SoftPciRst**. Read; Write-1-to-clear. Cold reset: 0.
16	**UserRst**. Read; Write-1-to-clear. Cold reset: 0.
15:14	**PmeTurnOffTime**. Read-write. Cold reset: 0. Bits Definition 00b 1 ms 01b 2 ms 10b 4 ms 11b 8 ms

13	**DisSbToNbPG**. Read-write. Cold reset: 0. 1=Disable NBPwrGood.
12	**DisableLdtPwrGood**. Read-write. Cold reset: 0. This is a LdtPwrGood control bit. 1=Disable the LdtPwrGood assertion along with NBPwrGood.
11	**SlpS3ToLdtPwrGdEn**. Read-write. Cold reset: 1. 1=De-assert LDT_PWRGD as long as SLP_S3# goes low.
10	**PwrGdDwnBeforeSlpS3**. Read-write. Cold reset: 0. 1=Delay SLP_S3 by 64 us and also qualify the FCH PwrGood with SLP_S3; This allows internal logic to put signals into correct states before turning off the S0 power.
9:6	Reserved.
5	**ShutDownFan0**. Read; Write-1-to-clear. Cold reset: 0.
4	**RemotePowerDownFromASF**. Read; Write-1-to-clear. Cold reset: 0.
3	**ThermalTripFromTemp**. Read; Write-1-to-clear. Cold reset: 0.
2	**Shutdown**. Read; Write-1-to-clear. Cold reset: 0.
1	**FourSecondPwrBtn**. Read; Write-1-to-clear. Cold reset: 0.
0	**ThermalTrip**. Read; Write-1-to-clear. This bit remains persistent though warm and cold resets and is reset on a G3->S5 transistion.

PMxC4 ResetCommand

Bits	Description
7	**ResetEn**. Read-write. Cold reset: 0. 0=Writing to bit Reset is not allowed. 1=Writing to bit Reset is allowed.
6	**ResetAllAcpi**. Write-1-only; Cleared-by-hardware. Cold reset: 0. Writing 1 to this bit emulates a reset button event.
5	**ResetButtonEn**. Read-write. Cold reset: 1. 1=Enable ResetButton. Sys_Reset_L assertion triggers reset. 0=Disable ResetButton. Set to 0 if the System_rst_L pin is not used or configured to Gevent19/gpio73 (function 1) in IOMUX 73h.
4	**ResetPcie**. Read-write. Cold reset: 0. 1=Reset the GPP ports. 0=Release the PCIe® reset.
3	**UsrRst2Pll**. Read-write. Cold reset: 1. 1=Stop PLL when the reset button is pressed.
2	**SelectDebug**. Read-write. Cold reset: 0. 0=Select PMxC0 [S5/Reset Status] to be S5/Reset Status register. 1=Select PMxC0 [S5/Reset Status] to be a debug status register.
1	**MemRstDisable**. Read-write. Cold reset: 0. 1=The memory reset function at DDR_RST# pin is disabled.
0	**Reset**. Write-1-only; Cleared-by-hardware. Cold reset: 0. Writing 1 to this bit causes a PCI reset. This bit is enabled by RestEn bit.

PMxC5 CF9 Shadow

Bits	Description
7:0	Alias of IOCF9.

PMxC8 Misc

Bits	Description
31:24	**ClkIntrVectorOrd**. Read-write. Cold reset: 0. Specifies the value used to identify the clock interrupt.
23	Reserved.
22	**ClkIntrVectorOrdEn**. Read-write. Cold reset: 0. 1=The system timer interrupt in the IOAPIC is tagged with a value defined by ClkIntrVectorOrd.
21	Reserved.
20	**ProcHotStsEn**. Read-write. Cold reset: 0. 1=Enable PROCHOT# to generate TwarnStatus SMIx84[16] and thermal throttle.
19	**UseCpuRst**. Read-write. Cold reset: 1. 0=System reset causes INIT# instead of CPURST#.
18	**UseBypassRom**. Read-write. Cold reset: 0. 1=Override the ROM straps and use BypassRomSel to determine which type of ROM to use. This is for BIOS debugging purposes or for systems having multiple BIOSes on board.
17:16	**BypassRomSel**. Read-write. Cold reset: 0. These two bits override the two ROM strap pins if [UseBypassRom] == 1. Bits — Definition 00b — LPC ROM 01b — Reserved 10b — Reserved 11b — SPI ROM
15	**HideSmbus**. Read-write. Cold reset: 0. 1=Hide the SMBus PCI configuration space and promote LPC bridge PCI configuration space to function 0.
14	Reserved.
13	**IdChangeEn**. Read-write. Cold reset: 0. 1=Allow the software to change D14F0x00[DeviceID] and D14F0x08[RevisionID].
12	**S5ResetOverride**. Read-write. Cold reset: 0. 1=Mask off internet PCI reset used in ACPI.
11	**WriteBackEnable**. Read-write. Cold reset: 0. 1=The WakeOnRing status bit is written back to HD audio controller upon system power up.
10	**LlbEn**. Read-write. Cold reset: 0. 1=LLB function is enabled and system won't wake up from ACPI S-state until LLB# is de-asserted.
9:8	**TempPolarity**. Read-write. Cold reset: 0. Temperature polarity control for THRMTRIP and TALERT respectively. 0=Active low. 1=Active high.
7	**DisablePciRom**. Read-write. Cold reset: 0. 1=Disable PCI from strap.
6	**TwarnEn**. Read-write. Cold reset: 0. 1=Enable TALERT# pin.
5	**EPromDeSelect**. Read-write. Cold reset: 0.
4	**UseAcpiStraps**. Read-write. Cold reset: 0.
3	**TDeadEn**. Read-write. Cold reset: 1. 1=GEVENT2 takes up the THRMTRIP function; When THRMTRIP pin is low and [TFatalEn] is set, hardware switches the system to S5 automatically.
2	**TFatalEn**. Read-write. Cold reset: 1. 1=Enable both the soft PciRst and the THRMTRIP function.
1	Reserved.
0	**CpuIoPullDownDrvStrength**. Read-write. Cold reset: 0. 1=The integrated pull-down drive strength of all CPU IOs are increased by 50%.

PMxCC IoDrvSth

Bits	Description
31:30	Reserved.
29:27	**IoDrvSth_Misc**. Read-write. Cold reset: 011b. IO drive strength for GA20, KBRST#, SERIRQ, and SATA_ACT# pads.
26:24	**IoDrvSth_Gpio**. Read-write. Cold reset: 011b. IO drive strength for GPIO3, and GPIO[48:52] pads.
23:21	**IoDrvSth_Gpio**. Read-write. Cold reset: 011b. IO drive strength for BMREQ#, GPIO[0, 2, 4, 5, 7, 8, 9, 13, 37, 38, 39, 40], GPOC[0, 1] pads.
20:18	**IoDrvSth_Req**. Read-write. Cold reset: 011b. IO drive strength for REQ[3:0]# when they are configured as GPIO.
17:15	Reserved.
14:12	**IoDrvSth_LPC**. Read-write. Cold reset: 011b. IO drive strength for LPC LAD, LFRAME# pads. The recommended setting for single load is 111b.
11:0	Reserved.

PMxD0 RstCntrl

Bits	Description
7:6	**RstLengthSel**. Read-write. Cold reset: 0. This field selects which register is defined at PMxD1.
5:0	Reserved.

PMxD1 Reset Function

Address: PMxD0[RstLengthSel].

Bits	Description
7:0	**RstFunctLength**.

PMxD1_x0 RstLength

Bits	Description
7:0	**RstLength**. Read-write. Cold reset: F4h. Specifies the A_RST# length. The amount of reset time = RstLength * 4096 * 69.84 ns.

PMxD1_x1 APURstLength

Bits	Description
7:0	**APURstLength**. Read-write. Cold reset: FBh. Specifies the APU_RST# length. The amount of reset time = ApuRstLength * 4096 * 69.84 ns.

PMxD1_x3 APUPwrGdLength

Bits	Description
7:0	**APUPwrGdLength**. Read-write. Cold reset: EAh. Specifies the LdtPwrGd latency. The amount of delay = APUPwrGdLength * 4096 * 69.84 ns.

PMxD2 Pmio

Bits	Description
7:6	Reserved.
5:4	**LpcClkDrvSth**. Read-write. Cold reset: 11b. Drive strength control for LpcClk[1:0] respectively. 0=Clock output is 4 mA. 1=Clock output is 8 mA.
3:0	Reserved.

PMxD6 IMC Gating

Bits	Description
15	**ImcEnable**. Read-write. Reset: 0. BIOS: See 2.17.7. 1=Enable IMC.
14:0	Reserved.

PMxD8 Eprom Index

Bits	Description
7:0	**EpromIndex**. Read-write. Cold reset: 0.

PMxD9 Eprom Data

Bits	Description
7:0	**EpromData**. Read-write. Cold reset: 0.

PMxDA SataConfig

Bits	Description
15:8	Reserved.
7:6	**RefDivSel**. Read-write. Cold reset: 10b. BIOS: See 2.17.3.5. This specifies the reference clock divider setting. Bits — Definition 00b — Divide by 1 (25 MHz or 48 MHz reference clock) 01b — Divide by 2 10b — Divide by 4 (100 MHz reference clock) 11b — Divide by 4 (100 MHz reference clock)

5:4	**RefClkSel**. Read-write. Cold reset: 01b. BIOS: See 2.17.3.5. This specifies the reference clock source selection for SATA PLL. Bits Definition 00b Reference clock from crystal oscillator via PAD_XTALI and PAD_XTALO. 01b Reference clock from internal clock through CP_PLL_REFCLK_P and CP_PLL_REFCLK_N via RDL. 10b Reserved. 11b Reserved.
3	Reserved.
2	**SetMaxGen2**. Read-write. Cold reset: 0. 0=SATA controller operates in maximum Gen3 (6.0 Gb/s) speed. 1=SATA controller operates in maximum Gen2 (3.0 Gb/s) speed for reduced power consumption.
1	Reserved.
0	**SataEnable**. Read-write. Cold reset: 1. BIOS: 1. 0=SATA controller is disabled. 1=SATA controller is enabled. Whenever SATA controller is being enabled by programming this field from 0 to 1, a IOCF9 software reset is recommended to reset the controller to a proper operational state.

PMxDC SataConfig2

Bits	Description
31:24	Reserved.
23	**BG_PWR_ON_OPTION**. Read-write. Reset: 0. BIOS: 1. This bit is to enable BANDGAP power on. 0=BG PWR ON control from SATA controller. 1=BG PWR is always powered ON.
22:17	Reserved.
16	**PllCalibEn: SATA PHY PLL calibration enable**. Read-write. Reset: 0. BIOS: See 2.17.3.3.
15:0	Reserved.

PMxE0 ABRegBar

Bits	Description
31:16	Reserved.
15:3	**ABRegBar**. Read-write. Cold reset: 0. BIOS: See 2.17.9. IO Base address of AB Configuration Registers. See AB configuration registers defined in 3.26.2 [AB Configuration Registers].
2:0	Reserved.

PMxE4 AB Misc Control

Bits	Description
31:2	Reserved.
1:0	**BlinkControl**. Read-write. Cold reset: 0. This field controls the BLINK pin behavior. BLINK pin can be used to control the on/off of a LED on the board. Bits Definition 00b Disabled. BLINK pin can be controlled by GPIO logic if so configured. 01b BLINK pin is repeatedly driven to low for 1 sec and then released for 3 sec. 10b BLINK pin is repeatedly driven to low for 2 sec and then released for 2 sec. 11b BLINK pin is always driven to low.

PMxE8 SDFlashCntrl

Bits	Description
7:1	Reserved.
0	**SDFlashEnable**. Read-write. Cold reset: 0. 0=Disable SD flash controller. 1=Enable SD flash controller, and GPIO[73:80] becomes the SD flash interface.

PMxEB AzEn

Bits	Description
7:0	Reserved.

PMxEC LpcGating

Bits	Description
7:3	Reserved.
2	**AbNoBypassEn**. Read-write. Cold reset: 0. BIOS: 1. 1=Tells the A-Link that the LPC cycle should not be bypassed when a retry has timed out.
1	**LpcA20En**. Read-write. Cold reset: 0. 1=Enable A20# input.
0	**LpcEnable**. Read-write. Cold reset: 1. 1=Enable LPC bridge.

PMxED USB Gating

Bits	Description
7:5	Reserved.
4	**UsbSmiEn**. Read-write. Cold reset: 0. 1=Enable SMI for USB legacy support.
3:2	Reserved.
1	**UsbIrqEn**. Read-write. Cold reset: 0. 1=Enable IRQ1/IRQ12 for USB legacy support.
0	Reserved.

PMxEF USB Enable

BIOS: See 2.17.2.3 [Enabling the xHCI Controller].

Bits	Description
7	**PortRoutingSelect**. Read-write. Cold reset: 1. Specifies controller(s) for USB ports 8 and 9. 0=EHCI 3 and OHCI 3 controllers. 1=xHCI controller.
6	Reserved.
5	**Usb3EhciEnable**. Read-write. Cold reset: 1. 1=Enable EHCI 3 controller (device 16h, function 0).
4	Reserved.
3	**Usb2EhciEnable**. Read-write. Cold reset: 1. 1=Enable EHCI 2 controller (device 13h, function 0).
2	Reserved.
1	**Usb1EhciEnable**. Read-write. Cold reset: 1. 1=Enable EHCI 1 controller (device 12h, function 0).
0	Reserved.

PMxF0 USB Control

Table 283: BIOS Recommendations for UsbPhyS5PwrDwnEnable

Option	Description	Setting	Comment
1	USB Wake from S5 not supported on the platform.	1	When the USB power rails USB PHY PLL, USB PHY core power and USB PHY DLL are connected to S0-S3 power, set the bit to 1 to disable the USB S4/S5 wake up function.
2	USB Wake from S5 supported on the platform.	0	When the USB power rails USB PHY PLL, USB PHY core power and USB PHY DLL are connected to S5 power, set the bit to 0 to enable the USB S4/S5 wakeup function.

Bits	Description
15	**PmioEhciMemSlpDis**. Read-write. Cold Reset: 0. 0=Enable EHCI memory sleep. 1=Disable EHCI memory sleep.
14:13	Reserved.
12	**Usb2BlGlobalClkGateEn**. Read-write. Cold reset: 1. 1=Enable USB2.0 B-Link Global Clock Gating.
11	Reserved.

10:8	**UsbSleepCtrl**. Read-write. Cold reset: 3h. Control on USB advanced asynchronous sleep function. Setting of 000b:100b are for the advanced asynchronous sleep. Bits — Definition 000b — Standard 10 us sleep. 001b — Advanced sleep for up to 2 micro frames. 010b — Advanced sleep for up to 4 micro frames. 011b — Advanced sleep for up to 6 micro frames. 100b — Advanced sleep for up to next micro frame. 101b — Reserved. 110b — Reserved. 111b — If CPU is in C-state and the controller has already exhausted the link list, it can simply stop the asynchronous packets until CPU resumes back to C0 state. In this case, the controller resumes back to its standard mode.
7:4	Reserved.
3	**UsbS5ResetEnable**. Read-write. Cold reset: 1. 1=Enable USB reset on S4/S5 resume detection.
2	**Mem_access_during_D3_enable**. Read-write. Cold reset: 1. BIOS: 0. 1=Enable memory space access during D3 state. 0=Disable memory space access during D3 state.
1	Reserved.
0	**UsbPhyS5PwrDwnEnable**. Read-write. Cold reset: 0. BIOS: See Table 283 [BIOS Recommendations for UsbPhyS5PwrDwnEnable]. 1=Enable S4/S5 USB PHY power down support and disable S4 USB wake up support. 0=Disable S4/S5 USB PHY power down support and enable S4 USB wakeup support. The bit has to be set to 0 to support S4 USB wake up.

3.26.13 Power Management Block 2 (PM2) Registers

PM2 register space is accessed through two methods:
- Indirect IO access through index/data address pair at IOCD0 [PM2_Index] and IOCD1 [PM2_Data]. Software first programs the offset into the index register IOCD0 [PM2_Index] and then reads/writes to/from the data register IOCD1 [PM2_Data].
- Memory mapped access through the AcpiMmio region. The ACPI registers range from FED8_0000h+400h to FED8_0000h+4FFh. See PMx04[MmioEn] for details on the AcpiMmio region.

PM2x00 Fan0InputControl

Bits	Description
7:4	Reserved.
3	**TwoRampAlgorithmEn**. Read-write. Reset: 0. 1=The two ramp fan control algorithm is enabled. 0=Disabled.
2:0	**FanInputControl**. Read-write. Reset: 101b. Bits — Definition 100b-000b — Reserved. 101b — FanOut0 is enabled and temperature input is from TempTsi, see PM2x8A, PM2x8B. 110b — FanOut0 is enabled and temperature input is 0. 111b — FanOut0 is disabled.

PM2x01 Fan0Control

When [AutoMode] == 1, the active duty cycle is controlled by the hardware automatically.
- If ActualTemperature < LowTemp{PM2x07, PM2x06}, DutyCycle = 0.
- If LowTemp{PM2x07, PM2x06} <= ActualTemperature < MedTemp{PM2x09, PM2x08}, DutyCycle = PM2x03 [LowDuty0].
- If MedTemp{PM2x09, PM2x08} <= ActualTemperature < HighTemp{PM2x0B, PM2x0A}:
 - If [LinearMode] == 0, DutyCycle = PM2x04 [MedDuty0].
 - If [LinearMode] == 1, DutyCycle = ((ActualTemperature - LowTemp{PM2x07, PM2x06}) * (PM2x05[Multiplier] + 1) >> PM2x05[DutySel]) + PM2x03 [LowDuty0].
- If ActualTemperature >= HighTemp{PM2x0B, PM2x0A}, DutyCycle = 100%.

In Automode, hysteresis limit defined by PM2x0C [LinearRange0] is applied to keep the fan from oscillating erratically.

Bits	Description
7:3	**LinearAdjust**. Read-write. Reset: 0. Specifies the additional offset to effective duty cycle under Linear mode.
2	**FanPolarity**. Read-write. Reset: 0. 0=FanOut0 drives low. 1=FanOut0 drives high.
1	**LinearMode**. Read-write. Reset: 0. 0=Use step function. 1=Use Linear function. See above description for details.
0	**AutoMode**. Read-write. Reset: 0. 0=FanOut0 is controlled by PM2x03 [LowDuty0]. 1=FanOut0 is controlled by the temperature input. See the above description for details on AutoMode.

PM2x02 Fan0Freq

Bits	Description
7:0	**FanFreq**. Read-write. Reset: 0. Normally, 4-wire fan runs at 25 kHz and 3-wire fan runs at 100 Hz. FanOut0 frequency is programmed as below:

Bits	Definition	Bits	Definition
00h	28.64 kHz	F8h	87 Hz
01h	25.78 kHz	F9h	58 Hz
02h	23.44 kHz	FAh	44 Hz
03h	21.48 kHz	FBh	35 Hz
04h	19.83 kHz	FCh	29 Hz
05h	18.41 kHz	FDh	22 Hz
F6h-06h	1/(FanFreq*2048*15 ns)	FEh	14 Hz
F7h	100 Hz	FFh	11 Hz

PM2x03 LowDuty0

Bits	Description
7:0	**LowDuty0**. Read-write. Reset: 0. Specifies Fan0 duty number if (LowTemp{PM2x07, PM2x06} <= temperature < MedTemp{PM2x09, PM2x08}). There are 256 time slots in one fan cycle. Duty number N represents the (N+1)th time slot. Fan actively spins in time slot0 ~ slotN, and stops from slot(N+1) ~ slot255. Bits — Definition 00h — Always stop. FEh-01h — Fan spins for time slots <= LowDuty, stops for time slots > LowDuty. FFh — Full speed run.

PM2x04 MedDuty0

Bits	Description
7:0	**MedDuty**. Read-write. Reset: 0. Specifies Fan0 duty number using step function if (MedTemp{PM2x09, PM2x08} <= temperature <= HighTemp{PM2x0B, PM2x0A}). There are 256 time slots in one fan cycle. Duty number N represents the (N+1)th time slot. Fan actively spins in time slot0 ~ slotN, and stops from slot(N+1) ~ slot255. Bits — Definition 00h — Always stop. FEh-01h — Fan spins for time slots <= MedDuty, stops for time slots > MedDuty. FFh — Full speed run.

PM2x05 Multiplier0

When (PM2x01[AutoMode] == 1) && (PM2x00[TwoRampAlgorithm] == 1), Fan0 is in TwoRamp mode. The slope value of ramp1 and ramp0 is: Slope[7:0] = {PM2x05[DutySel], PM2x05[Multiplier]}. Slope[7:2] are the integer bits and Slope[1:0] are the fractional bits of the ramp slope.
BIOS has to calculate Slope[7:0] using the following equation:
Slope = (FF00h - {LowDuty0[7:0],0000_0000b}) / (HighTemp0[15:0] - Hysteresis0[15:0] - MedTemp0[15:0])
For example, if our setting is:

LowDuty0[7:0] = 25% = 40h
HighTemp0[15:0] = 90°C = 5A00h
MedTemp0[15:0] = 40°C = 2800h
Hysteresis0[15:0] = 10°C = 0A00h

Then:

Slope = (FF00h - 4000h) / (5A00h - 0A00h - 2800h) = 48896 / 10240 = 4.775.

Convert the number 4.775 into our 8-bit format: 0001_0011b (13h). BIOS should program the Multiplier0 register as 13h in this example.

Bits	Description
7:6	**DutySel**. Read-write. Reset: 0. When Fan0 is programmed in AutoMode with linear function being selected, this field selects part of duty to be fed into fan as described in PM2x01 [Fan0Control] register description. When Fan0 is programmed in TwoRamp mode, this field specifies the upper 2 bits in Slope[7:0]. See the above register description for detail.
5:0	**Multiplier**. Read-write. Reset: 0. When Fan0 is programmed in AutoMode with linear function being selected, this field specifies the factor to calculate duty number as described in PM2x01 [Fan0Control] register description. When Fan0 is programmed in TwoRamp mode, this field specifies the lower 6 bits in Slope[7:0]. See the above register description for detail.

PM2x06 LowTemp0Lo

Bits	Description
7:0	**LowTemp0[7:0]**. Read-write. Reset: 0. Specifies the lower 8 bits of low temperature threshold.

PM2x07 LowTemp0Hi

Bits	Description
7:0	**LowTemp0[15:8]**. Read-write. Reset: 0. Specifies the higher 8 bits of low temperature threshold.

PM2x08 MedTemp0Lo

Bits	Description
7:0	**MedTemp0[7:0]**. Read-write. Reset: 0. Specifies the lower 8 bits of medium temperature threshold.

PM2x09 MedTemp0Hi

Bits	Description
7:0	**MedTemp0[15:8]**. Read-write. Reset: 0. Specifies the higher 8 bits of medium temperature threshold.

PM2x0A HighTemp0Lo

Bits	Description
7:0	**HighTemp0[7:0]**. Read-write. Reset: 0. Specifies the lower 8 bits of high temperature threshold.

PM2x0B HighTemp0Hi

Bits	Description
7:0	**HighTemp0[15:8]**. Read-write. Reset: 0. Specifies the higher 8 bits of high temperature threshold.

PM2x0C LinearRange0

Bits	Description
7:0	**LinearRange**. Read-write. Reset: 0. Specifies a variable range that Fan0 can tolerate. Fan0 is not affected if temperature varies within this range.

PM2x0D LinearHoldCount0

Bits	Description
7:0	**LinearHoldCount**. Read-write. Reset: 0. Specifies the fan cycles to be waited before duty cycle can be changed.

PM2x0E Fan0Hysteresis

Bits	Description
15:8	**HysteresisHi**. Read-write. Reset: 0. Specifies the hysteresis value (in temperature) of the Two Ramp Fan Control Algorithm. The unit is °C. See PM2x05 [Multiplier0] on how to program this register.
7:0	**HysteresisLo**. Read-write. Reset: 0. This byte should always be programmed as 0.

PM2x50 Med2Temp0Lo

Bits	Description
7:0	**Med2Temp0[7:0]**. Read-write. Reset: 0. IF (PM2x00[TwoRampAlgorithmEn] == 1) THEN It specifies the lower byte of the temperature value of the turning point on the ramp. ELSE Unused. ENDIF.

PM2x51 Med2Temp0Hi

Bits	Description
7:0	**Med2Temp0[15:8]**. Read-write. Reset: 0. IF (PM2x00[TwoRampAlgorithmEn] == 1) THEN It specifies the higher byte of the temperature value of the turning point on the ramp. ELSE Unused. ENDIF.

PM2x52 Med2Duty0

Bits	Description
7:0	**Med2Duty0[7:0]**. Read-write. Reset: 0. IF (PM2x00[TwoRampAlgorithmEn] == 1) THEN It specifies the fan duty value of the turning point on the ramp. ELSE Unused. ENDIF.

PM2x53 Multiplier2_0

BIOS has to calculate Multiplier2_0[7:0] using the following equation:
Multiplier2_0=({Med2Duty0[7:0], 0000_0000b}-{LowDuty0[7:0], 0000_0000b}) / (Med2Temp0[15:0]-MedTemp0[15:0]).
For example, if our setting is:

PM2x52[Med2Duty0[7:0]] = 50% = 80h
PM2x03[LowDuty0[7:0]] = 25% = 40h
Med2Temp0[15:0] = {PM2x51, PM2x50} = 70°C = 4600h
MedTemp0[15:0] = {PM2x09, PM2x08} = 40°C = 2800h
Then:
Multiplier2_0[7:0] = (8000h-4000h) / (4600h-2800h) = 16384 / 7680 = 2.133.
Convert the number 4.775 into our 8bit format: 0000_1000b (08h). BIOS should program Multiplier2_0 register as 08h in this example.

Bits	Description
7:0	**Multiplier2_0[7:0]**. Read-write. Reset: 0. BIOS: See above description. IF (PM2x00[TwoRampAlgorithmEn] == 1) THEN It specifies the slope value of ramp1lo and ramp0lo. Bit — Definition [1:0] — Integer bits of the ramp slope. [7:2] — Fractional bits of the ramp slope.

PM2x60 FanStatus

Bits	Description
7:1	Reserved.
0	**Fan0SpeedTooSlow**. Read; Write-1-to-clear. Reset: 0. 1=Fan0 runs slower than the value in the Fan0SpeedLimit{PM2x68, PM2x67}.

PM2x61 FanINTRouteLo

Bits	Description
7:2	Reserved.
1:0	**Fan0INTRoute**. Read-write. Reset: 0. Bits — Definition 00b — No SCI/SMI generated 01b — SMI 10b — SMI or SCI depending on GEVENT13 routing 11b — No SCI/SMI generated

PM2x63 SampleFreqDiv

Bits	Description
7:4	**LinearRangeOutLimit[4:1]**. Read-write. Reset:0. LinearRangeOutLimit[7:0] = {000b, LinearRangeOutLimit[4:1], 1b}. LinearRangeOutLimit specifies how close the fan duty follows the target duty cycle and is only used when the fan duty is changing. It is different from the PM2x0C [LinearRange0], which works like a hysteresis and used when fan duty is not changing.
3	Reserved.

2	**FanLinearEnhanceEn2**. Read-write. Reset: 0. 1=The positive hysteresis of fan duty is removed; PM2x0C [LinearRange0] only applies to the negative direction; As a result, the fan duty increases once the temperature is increased instead of waiting for a hysteresis.
1:0	**SampleFreqDiv**. Read-write. Reset: 0. This field specifies the sampling rate of Fan Speed. Bits Definition 00b Base (22.5 kHz) 01b Base (22.5 kHz) / 2 10b Base (22.5 kHz) / 4 11b Base (22.5 kHz) / 8

PM2x64 FanDebounceCounterLo

Bits	Description
7:0	**FanDebounceCounter[7:0]**. Read-write. Reset: 0. Specifies the lower 8 bits of the debounced counter when measuring fan speed.

PM2x65 FanDebounceCounterHi

Bits	Description
7:0	**FanDebounceCounter[15:8]**. Read-write. Reset: 0. Specifies the high 8 bits of the debounced counter when measuring fan speed.

PM2x66 Fan0DetectorControl

Bits	Description
7:5	Reserved.
4	**ShutDownEnable**. Read-write. Reset: 0. 1=The system will be shut down if PM2x60[Fan0SpeedTooSlow] remains 1 for more than 4 seconds.
3:2	Reserved.
1	**UseAverage**. Read-write. Reset: 0. 0=Do not to use average fan0 speed. 1=Use average fan0 speed.
0	**FanDetectorEnable**. Read-write. Reset: 0. 0=Disable fan0 speed measurement. 1=Enable fan0 speed measurement.

PM2x67 Fan0SpeedLimitLo

Bits	Description
7:0	**FanSpeedLimit[7:0]**. Read-write. Reset: 0. Specifies the lower 8 bits of Fan0SpeedLimit.

PM2x68 Fan0SpeedLimitHi

Bits	Description
7:0	**FanSpeedLimit[15:8]**. Read-write. Reset: 0. Specifies the higher 8 bits of Fan0SpeedLimit.

PM2x69 Fan0SpeedLo

Bits	Description
7:0	**FanSpeed[7:0]**. Read-only; Updated-by-hardware. Reset: 0. Specifies the lower 8 bits of fan0 speed.

PM2x6A Fan0SpeedHi

Bits	Description
7:0	**FanSpeed[15:8]**. Read-only; Updated-by-hardware. Reset: 0. Specifies the higher 8 bits of fan0 speed.

PM2x8A TempTsiLo

Bits	Description
7:0	**TempTsi[7:0]**. IF **(**PM2x8F [TempTsiWe] == 1) THEN Read-Write. ELSE Read-only; updated-by-hardware. ENDIF. Reset: 0. Specifies the lower 8 bits of TempTsi. TempTsi = {PM2x8B, PM2x8A}.

PM2x8B TempTsiHi

Bits	Description
7:0	**TempTsi[15:8]**. IF **(**PM2x8F [TempTsiWe] == 1) THEN Read-Write. ELSE Read-only; updated-by-hardware. ENDIF. Reset: 0. Specifies the higher 8 bits of TempTsi. TempTsi = {PM2x8B, PM2x8A}.

PM2x8C TempTsiLimitLo

Bits	Description
7:0	**TempTsiLimit[7:0]**. Read-write. Reset: 0. Specifies the lower 8 bits of TempTsiLimit. TempTsiLimit = {PM2x8D, PM2x8C}.

PM2x8D TempTsiLimitHi

Bits	Description
7:0	**TempTsiLimit[15:8]**. Read-write. Reset: 0. Specifies the higher 8 bits of TempTsiLimit. TempTsiLimit = {PM2x8D, PM2x8C}.

PM2x8E TempTsiChangeLimit

If (TempTsiChangeLimit != 0), filtering is applied to TempTsi{PM2x8B [TempTsiHi], PM2x8A [TempTsiLo]} as below:

- If TempTsi_New > (TempTsi_Old + (TempTsiChangeLimit<<6)), then ...(action to take)

- If TempTsi_New < (TempTsi_Old - (TempTsiChangeLimit<<6)), then ...(action to take)

- If (Temp_Old - (TempTsiChangLimit<<6)) <= TempTsi_New <= (TempTsi_Old + (TempChange-Limit<<6)), then ...(action to take)

Bits	Description
7:0	**TempTsiChangeLimit**. Read-write. Reset: 0.

PM2x8F TempTsiWe

Bits	Description
7:6	Reserved.
5	**TempTsiWe: TempTsi Write Enable**. Read-write. Reset: 0. 0=TempTsi registers, PM2x8B [TempTsiHi] and PM2x8A [TempTsiLo], are read-only and updated by hardware with the result from TempTsi sensor. 1=TempTsi registers are writable only by host and IMC; they are not updated by hardware with the result from the TempTsi sensor.
4:0	Reserved.

PM2x90 TempTsiStatus

Bits	Description
7:6	Reserved.
5	**TempTsiStatus**. Read; Write-1-to-clear. Reset: 0. 1=TempTsi{PM2x8B [TempTsiHi], PM2x8A [TempTsiLo]} is out of the limit.
4:0	Reserved.

PM2x92 TempTsiControl

Bits	Description
7:4	Reserved.
3:2	**TempTsiControl**. Read-write. Reset: 0. TempTsi sensor is enabled if (TempTsiControl != 0). Bits — Definition 00b — Disable. 01b — If (TempTsi > TempTsiLimit), then PM2x90[TempTsiStatus] = 1. 10b — If (TempTsi < TempTsiLimit), then PM2x90[TempTsiStatus] = 1. 11b — If ((TempTsiHi > TempTsiLimitLo) \|\| (TempTsiHi < TempTsiLimitHi)), then PM2x90[TempTsiStatus] = 1.
1:0	Reserved.

PM2x94 TempTsiINTRoute

Bits	Description
7:4	Reserved.

3:2	**TempTsiINTRoute**. Read-write. Reset: 0. Bits Definition 00b No SCI/SMI generated. 01b SMI. 10b SMI or SCI according to GEVENT 13 INT routing. 11b No SCI/SMI generated.
1:0	Reserved.

PM2xDF TempTsiRstSel

Bits	Description
7:6	Reserved.
5	**TempTsiRstSel**. Read-write. Reset: 0. 1=Thermal diode monitoring function is not stopped by reset.
4:0	Reserved.

PM2xE0 AlertThermaltripStatus

Bits	Description
7:2	Reserved.
1	**ThermalTripStatus**. Read-only; Updated-by-hardware. Reset: 0. 0=Current temperature is not above ThermalTripLimit ({PM2xE4, PM2xE3}). 1=Current temperature is above ThermalTripLimit ({PM2xE4, PM2xE3}).
0	**AlertStatus**. Read-only; Updated-by-hardware. Reset: 0. 0=Current temperature is not above Alert-Limit. 1=Current temperature is above AlertLimit.

PM2xE1 AlertLimitLo

Bits	Description
7:0	**AlertLimit[7:0]**. Read-write. Reset: 0.

PM2xE2 AlertLimitHi

Bits	Description
7:0	**AlertLimit[15:8]**. Read-write. Reset: 0.

PM2xE3 ThermalTripLimitLo

Bits	Description
7:0	**ThermalTripLimit[7:0]**. Read-write. Reset: 0. ThermalTripLimit = {PM2xE4, PM2xE3}.

PM2xE4 ThermalTripLimitHi

Bits	Description
7:0	**ThermalTripLimit[15:8]**. Read-write. Reset: 0. ThermalTripLimit = {PM2xE4, PM2xE3}.

PM2xE5 AlertThermaltripControl

Bits	Description
7:5	**TempSelAlert**. Read-write. Reset: 0. This field selects which temperature sensor is the event source. It converts the selected Temp input pin into either TAlert or ThermalTrip function. Bits — Field Definition 100b-000b — Reserved 101b — TempTsi 111b-110b — Reserved
4:2	Reserved.
1:0	**AlertControl**. Read-write. Reset: 0. Bit — Field Definition [1] — Enable ThermalTrip on the selected Temp input [0] — Enable TAlert on the selected Temp input

3.26.14 Standard ACPI Registers

These are the standard registers defined by the ACPI specification. In order to access these ACPI registers, PMx74[DecEnAcpi] must be programmed to 1. ACPI registers are accessed through two methods:

- IO mapped access through base addresses defined by PMx60 through PMx6E.
- Memory mapped access through the AcpiMmio region. The ACPI registers range from FED8_0000h+800h to FED8_0000h+8FFh. See PMx04[MmioEn] for details on the AcpiMmio region.

3.26.14.1 AcpiPmEvtBlk

The IO mapped base address of this register block is defined by PMx60.

AcpiPmEvtBlkx00 Pm1Status

Bits	Description
15	**WakeStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=The system is in the sleep state and a wake-up event occurs.
14	**PciExpWakeStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=The system wake is due to a PCI Express® wakeup event.
13:11	Reserved.
10	**RtcStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=RTC generates an alarm. If both AcpiPmEvtBlkx02[RtcEn] and this bit are set to 1, a power management event: SCI, SMI, or resume event, is generated.
9	Reserved.

8	**PwrBtnStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. Power button status bit. 1=The Power Button is pressed. In the system working state, if AcpiPmEvtBlkx02[PwrBtnEn] and this bit are both set to 1, an interrupt event is raised. In the sleeping or soft-off state, a wake event is generated when the power button is pressed regardless of the setting of AcpiPmEvtBlkx02[PwrBtnEn].
7:6	Reserved.
5	**GblStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=An SCI is generated due to the BIOS wanting the attention of the SCI handler; Writing 1 to PMx74[BiosRls] sets this bit.
4	**BmStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0.
3:1	Reserved.
0	**TmrStatus**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. Timer carry status bit. 1=The 31st bit of the 32-bit counter changes from low to high or high to low. If both AcpiPmEvtBlkx02[TmrEn] and this bit are set to 1, an interrupt event is raised.

AcpiPmEvtBlkx02 Pm1Enable

Bits	Description
15	Reserved.
14	**PciExpWakeDis**. Read-write. Reset: 1. 1=Disable the inputs to AcpiPmEvtBlkx00[PciExpWakeStatus] from waking the system.
13:11	Reserved.
10	**RtcEn**. Read-write. Reset: 0. 1=A wake event is generated whenever AcpiPmEvtBlkx00[RtcStatus] is also set to 1.
9	Reserved.
8	**PwrBtnEn**. Read-write. Reset: 0. 1=A power management event (SCI or wake) is generated whenever AcpiPmEvtBlkx00[PwrBtnStatus] is also set to 1.
7:6	Reserved.
5	**GblEn**. Read-write. Reset: 0. 1=An SCI is raised whenever AcpiPmEvtBlkx00[GblStatus] is also set to 1.
4:1	Reserved.
0	**TmrEn**. Read-write. Reset: 0. This is the timer carry interrupt enable bit. 1=An SCI event is generated whenever the AcpiPmEvtBlkx00[TmrStatus] is set to 1. 0=No interrupt is generated when the AcpiPmEvtBlkx00[TmrStatus] bit is set to 1.

3.26.14.2 AcpiPm1CntBlk

The IO mapped base address of this register block is defined by PMx62.

AcpiPm1CntBlkx00 PmControl

Bits	Description
15:14	Reserved.
13	**SlpTypeEn**. RAZ; Write-1-only. Reset: 0. 1=The system sequences into the sleeping state defined by SlpTyp when PMxBE [SlpTypeControl] is set to 1.

12:10	**SlpTyp**. Read-write. Reset: 0. Specifies the sleep state the system enters if SlpTypeEn is set to 1. This design currently implements 5 states: S0, S1, S3, S4, and S5.
9:3	Reserved.
2	**GblRls**. RAZ; Write-only. Reset: 0. If SMIxB0[17:16] is set to 01b, writing 1 to this bit generates SMI and sets SMIx88[8].
1	**BmRld**. Read-write. Reset: 0.
0	**SciEn**. Read-write. Reset: 0. Selects the power management event to be either an SCI or SMI interrupt for the power management events. 1=Power management events generate SCI interrupts. 0=Power management events generate SMI interrupts.

3.26.14.3 AcpiPm2CntBlk

The IO mapped base address of this register block is defined by PMx6E.

AcpiPm2CntBlkx00 Pm2Control

Bits	Description
7:1	Reserved.
0	**ArbDis**. Read-write. Reset: 0.

3.26.14.4 AcpiPmTmrBlk

The IO mapped base address of this register block is defined by PMx64.

AcpiPmTmrBlkx00 TmrValue/ETmrValue

Bits	Description
31:0	**TmrValue**. Read-only. Reset: X. It returns the running count of the power management timer (ACPI timer).

3.26.14.5 CpuCntBlk

The IO mapped base address of this register block is defined by PMx66.

CpuCntBlkx00 ClkValue

Bits	Description
31:5	Reserved.
4	**ThtEn**. Read-write. Reset: 0. 1=Enable clock throttling as programmed in ClkValue.
3:1	**ClkValue**. Read-write. Reset: 0. Specifies the throttle interval for STPCLK in software clock throttling. Bits Definition Bits Definition 000b 50% 100b 50% 001b 12.5% 101b 62.5% 010b 25% 110b 75% 011b 37.5% 111b 87.5%
0	Reserved.

CpuCntBlkx04 PLvl2

Bits	Description
7:0	**PLvl2**. RAZ; Read-only. Reset: 0. Reads to this register return all zeros and generate an “enter C2” sequence to APU.

CpuCntBlkx05 PLvl3

Bits	Description
7:0	**PLvl3**. RAZ; Read-only. Reset: 0. Reads to this register return all zeros and generate an “enter C3” sequence to APU.

3.26.14.6 AcpiGpe0Blk

The IO mapped base address of this register block is defined by PMx68.

AcpiGpe0Blkx00 EventStatus

Bits	Description
31:0	**EventStatus**. Read; Write-1-to-clear. Reset: X. Each bit represents an ACPI event status. For each bit, 1=The selected event input equals to the corresponding value in SMIx08 [SciTrig]. Configuration for the events are located at SMIx08 [SciTrig] through SMIx78 [SciMap14]. The status bits are also mirrored in SMIx00 [Event_Status].

AcpiGpe0Blkx04 EventEnable

Bits	Description
31:0	**EventEnable**. Read-write. Reset: 0. Each bit controls whether ACPI should generate wakeup and SCI interrupt. The enable bits are also mirrored in SMIx04 [Event_Enable].

3.26.14.7 SmiCmdBlk

The IO mapped base address of this register block is defined by PMx6A.

SmiCmdBlkx00 SmiCmdPort

Bits	Description
7:0	**SmiCmdPort**. Read-write. Reset: 0. When SMI command port is enabled, a write to this port generates SMI, NMI or IRQ3 depending on the setting of SMIxB0[23:22]. A read from this port returns the previously written value but does not generate SMI.

SmiCmdBlkx01 SmiCmdStatus

Bits	Description
7:0	**SmiCmdStatus**. Read-write. Reset: 0. Used by BIOS and OS.

3.26.15 SMI Registers

SMI register space is accessed through the AcpiMmio region. The SMI registers range from FED8_0000h+200h to FED8_0000h+2FFh. See PMx04[MmioEn].

SMIx00 Event_Status

Bits	Description
31:0	**EventStatus**. Read; Write-1-to-clear. Reset: X. This is a mirror register of the standard ACPI register AcpiGpe0Blkx00 [EventStatus]. Each event status bit is set when the selected event input equals to the corresponding value in SMIx08 [SciTrig].

SMIx04 Event_Enable

Bits	Description
31:0	**EventEnable**. Read-write. Reset: 0. This is the mirror register of the standard ACPI register AcpiGpe0Blkx04 [EventEnable]. Each bit controls whether ACPI should generate wake up and SCI interrupt.

SMIx08 SciTrig

Each bit in this register controls the way to set each corresponding bit in SMIx00 [Event_Status].

Bits	Description
31	**SciTrig31**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 31 (SMIx00[31]). 0=Active low. 1=Active high.
30	**SciTrig30**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 30 (SMIx00[30]). 0=Active low. 1=Active high.
29	**ScTrig29**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 29 (SMIx00[29]). 0=Active low. 1=Active high.
28	**SciTrig28**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 28 (SMIx00[28]). 0=Active low. 1=Active high.
27	**SciTrig27**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 27 (SMIx00[27]). 0=Active low. 1=Active high.
26	**SciTrig26**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 26 (SMIx00[26]). 0=Active low. 1=Active high.
25	**SciTrig25**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 25 (SMIx00[25]). 0=Active low. 1=Active high.
24	**SciTrig24**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 24 (SMIx00[24]). 0=Active low. 1=Active high.
23	**SciTrig23**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 23 (SMIx00[23]). 0=Active low. 1=Active high.
22	**SciTrig22**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 22 (SMIx00[22]). 0=Active low. 1=Active high.
21	**SciTrig21**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 21 (SMIx00[21]). 0=Active low. 1=Active high.

20	**SciTrig20**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 20 (SMIx00[20]). 0=Active low. 1=Active high.
19	**SciTrig19**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 19 (SMIx00[19]). 0=Active low. 1=Active high.
18	**SciTrig18**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 18 (SMIx00[18]). 0=Active low. 1=Active high.
17	**SciTrig17**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 17 (SMIx00[17]). 0=Active low. 1=Active high.
16	**SciTrig16**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 16 (SMIx00[16]). 0=Active low. 1=Active high.
15	**SciTrig15**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 15 (SMIx00[15]). 0=Active low. 1=Active high.
14	**SciTrig14**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 14 (SMIx00[14]). 0=Active low. 1=Active high.
13	**SciTrig13**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 13 (SMIx00[13]). 0=Active low. 1=Active high.
12	**SciTrig12**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 12 (SMIx00[12]). 0=Active low. 1=Active high.
11	**SciTrig11**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 11 (SMIx00[11]). 0=Falling edge. 1=Active high.
10	**SciTrig10**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 10 (SMIx00[10]). 0=Active low. 1=Active high.
9	**SciTrig9**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 9 (SMIx00[9]). 0=Active low. 1=Active high.
8	**SciTrig8**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 8 (SMIx00[8]). 0=Active low. 1=Active high.
7	**SciTrig7**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 7 (SMIx00[7]). 0=Active low. 1=Active high.
6	**SciTrig6**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 6 (SMIx00[6]). 0=Active low. 1=Active high.
5	**SciTrig5**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 5 (SMIx00[5]). 0=Active low. 1=Active high.
4	**SciTrig4**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 4 (SMIx00[4]). 0=Active low. 1=Active high.
3	**SciTrig3**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 3 (SMIx00[3]). 0=Active low. 1=Active high.
2	**SciTrig2**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 2 (SMIx00[2]). 0=Active low. 1=Active high.
1	**SciTrig1**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 1 (SMIx00[1]). 0=Active low. 1=Active high.
0	**SciTrig0**. Read-write. Reset: 1. The bit controls the way to set Event_Status bit 0 (SMIx00[0]). 0=Active low. 1=Active high.

SMIx0C SciLevl

Reset: 0000_0000h. This register specifies the trigger mode for each of the corresponding bit in SMIx00 [Event_Status].

Bits	Description
31	**SciLevl31**. Read-write. 0=Edge trigger. 1=Level trigger.
30	**SciLevl30**. Read-write. 0=Edge trigger. 1=Level trigger.
29	**SciLevl29**. Read-write. 0=Edge trigger. 1=Level trigger.
28	**SciLevl28**. Read-write. 0=Edge trigger. 1=Level trigger.
27	**SciLevl27**. Read-write. 0=Edge trigger. 1=Level trigger.
26	**SciLevl26**. Read-write. 0=Edge trigger. 1=Level trigger.
25	**SciLevl25**. Read-write. 0=Edge trigger. 1=Level trigger.
24	**SciLevl24**. Read-write. 0=Edge trigger. 1=Level trigger.
23	**SciLevl23**. Read-write. 0=Edge trigger. 1=Level trigger.
22	**SciLevl22**. Read-write. 0=Edge trigger. 1=Level trigger.
21	**SciLevl21**. Read-write. 0=Edge trigger. 1=Level trigger.
20	**SciLevl20**. Read-write. 0=Edge trigger. 1=Level trigger.
19	**SciLevl19**. Read-write. 0=Edge trigger. 1=Level trigger.
18	**SciLevl18**. Read-write. 0=Edge trigger. 1=Level trigger.
17	**SciLevl17**. Read-write. 0=Edge trigger. 1=Level trigger.
16	**SciLevl16**. Read-write. 0=Edge trigger. 1=Level trigger.
15	**SciLevl15**. Read-write. 0=Edge trigger. 1=Level trigger.
14	**SciLevl14**. Read-write. 0=Edge trigger. 1=Level trigger.
13	**SciLevl13**. Read-write. 0=Edge trigger. 1=Level trigger.
12	**SciLevl12**. Read-write. 0=Edge trigger. 1=Level trigger.
11	**SciLevl11**. Read-write. 0=Edge trigger. 1=Level trigger.
10	**SciLevl10**. Read-write. 0=Edge trigger. 1=Level trigger.
9	**SciLevl9**. Read-write. 0=Edge trigger. 1=Level trigger.
8	**SciLevl8**. Read-write. 0=Edge trigger. 1=Level trigger.
7	**SciLevl7**. Read-write. 0=Edge trigger. 1=Level trigger.
6	**SciLevl6**. Read-write. 0=Edge trigger. 1=Level trigger.
5	**SciLevl5**. Read-write. 0=Edge trigger. 1=Level trigger.
4	**SciLevl4**. Read-write. 0=Edge trigger. 1=Level trigger.
3	**SciLevl3**. Read-write. 0=Edge trigger. 1=Level trigger.
2	**SciLevl2**. Read-write. 0=Edge trigger. 1=Level trigger.
1	**SciLevl1**. Read-write. 0=Edge trigger. 1=Level trigger.
0	**SciLevl0**. Read-write. 0=Edge trigger. 1=Level trigger.

SMIx10 SmiSciStatus

Bits	Description
31:0	**SmiSciStatus**. Read; Write-1-to-clear. Reset: 0. Each bit indicates the corresponding SmiSci status. The input of each bit is controlled by the corresponding SMIx08 [SciTrig] bit. Note: this function can be considered as a superset of SMIx00 [Event_Status]. When one of the bits is set and its SMIx14 [SmiSciEn] is also set, it triggers a SMI to call the BIOS. After the BIOS has serviced the SMM and cleared its status, the internal logic automatically sets the corresponding SMIx00 [Event_Status] bit and thereby triggering a SCI.

SMIx14 SmiSciEn

Bits	Description
31:0	**SmiSciEn**. Read-write. Reset: 0. Each bit controls if SMI message is generated when the corresponding SMIx10 [SmiSciStatus] bit is set to 1. For each bit, 0=Not to send SMI message when the corresponding SMIx10 [SmiSciStatus] bit is set. 1=Send SMI message when the corresponding SMIx10 [SmiSciStatus] bit is set.

SMIx18 SwSciEn

Bits	Description
31:0	**SwSciEn**. Read-write. Reset: 0. This register is used as a software mechanism to trigger SCI. For each bit, 1=Software can write to SMIx1C [SwSciData] and set the corresponding SMIx00 [Event_Status] bit. Note: The setting of this bit needs to match with SMIx08 [SciTrig] and SMIx0C [SciLevl] in order to set the status bit.

SMIx1C SwSciData

Bits	Description
31:0	**SwSciData**. Read-write. Reset: 0. This is the software data path to set the corresponding SMIx00 [Event_Status] bit when SMIx18 [SwSciEn] is set.

SMIx20 SciSleepDisable

Bits	Description
31:0	**SciSleepDisable**. Read-write. Reset: 0. This register is used to ignore EVENT pins that are powered in the main power domain instead of auxiliary power domain. For each bit, 1=The corresponding SMIx00 [Event_Status] bit is masked off whenever the system goes to S3 or higher sleep state.

SMIx30 CapturedData

Bits	Description
31:0	**CapturedData**. Read-only. Reset: X. This is the buffer to capture write data for the last transaction that caused an SMI. Note: This buffer has no meaning for a read trap.

SMIx34 CapturedValid

Bits	Description
7:4	Reserved.
3:0	**CapturedValid**. Read-only. Reset: X. This is the byte valid buffer to signal which byte is captured for the last transaction that caused the SMI. Bit Definition [0] byte 0 [1] byte 1 [2] byte 2 [3] byte 3

SMIx38 EPBIF_AER_Straps

Bits	Description
31:28	Reserved.
27	**StrapBifInternalErrEnFch**. Read-write. Reset: 0. Internal error enable.
26	**StrapBifPoisonedAdvisoryNonfatalAFch**. Read-write. Reset: 1. Poisoned TLP as advisory nonfatal.
25	**StrapBifAcsDirectTranslatedP2pFch**. Read-write. Reset: 1. ACS direct translated P2P enable.
24	**StrapBifAcsUpstreamForwardingFch**. Read-write. Reset: 1. ACS upstream forwarding enable.
23	**StrapBifAcsP2pCompletionRedirectFch**. Read-write. Reset: 1. ACS P2P completion redirect enable.
22	**StrapBifAcsP2pRequestRedirectFch**. Read-write. Reset: 1. ACS P2P request redirect enable.
21	**StrapBifAcsTranslationBlockingFch**. Read-write. Reset: 1. ACS translation blocking enable.
20	**StrapBifAcsSourceValidationFch**. Read-write. Reset: 1. ACS source validation enable.
19	**StrapBifAcsEnFch**. Read-write. Reset: 1. ACS enable.
18	**StrapBifFirstRcvdErrLogFch**. Read-write. Reset: 1. First received error log.
17	**StrapBifEcrcCheckEnFch**. Read-write. Reset: 1. ECRC check enable.
16	**StrapBifEcrcGenEnFch**. Read-write. Reset: 0. ECRC generate enable.
15	**StrapBifCplAbortErrEnFch**. Read-write. Reset: 1. Completer abort error enable.
14	**StrapBifRxIgnoreVend0UrFch**. Read-write. Reset: 1. Ignore Vendor 0 error.
13	**StrapBifRxIgnoreTcErrFch**. Read-write. Reset: 1. Ignore traffic class error.
12	**StrapBifRxIgnoreMsgErrFch**. Read-write. Reset: 1. Ignore message error.
11	**StrapBifRxIgnoreMaxPayloadErrFch**. Read-write. Reset: 1. Ignore maximum payload error.

10	**StrapBifRxIgnoreLenMismatchErrFch**. Read-write. Reset: 1. Ignore length mismatch error.
9	**StrapBifRxIgnoreIoUrErrFch**. Read-write. Reset: 1. Ignore IO UR error.
8	**StrapBifRxIgnoreIoErrFch**. Read-write. Reset: 1. Ignore IO error.
7	**StrapBifRxIgnoreEpErrFch**. Read-write. Reset: 1. Ignore poisoned TLP error.
6	**StrapBifRxIgnoreCplErrFch**. Read-write. Reset: 1. Ignore completion error.
5	**StrapBifRxIgnoreCfgUrFch**. Read-write. Reset: 1. Ignore configuration UR error.
4	**StrapBifRxIgnoreCfgErrFch**. Read-write. Reset: 1. Ignore configuration error.
3	**StrapBifRxIgnoreBeErrFch**. Read-write. Reset: 1. Ignore byte enable error.
2	**StrapBifErrReportingDisFch**. Read-write. Reset: 1. Error reporting disable.
1	**StrapBifAerEnFch**. Read-write. Reset: 1. Advanced Error Reporting (AER) enable.
0	**StrapBifStickyOverrideS5**. Read-write. Reset: 0. 1=Values in this register would override straps loaded from EEPROM.

SMIx3C DataErrorStatus

Bits	Description
31:8	Reserved.
7	**SIrqIochk**. Read; Write-1-to-clear. Reset: 0. Serial IO check error.
6	**SataPerr**. Read; Write-1-to-clear. Reset: 0. BIOS: See 2.17.3.9. SATA controller internal parity error status.
5	**UmiUncorrectableErr**. Read; Write-1-to-clear. Reset: 0. UMI uncorrectable error status.
4	**UmiCorrectableErr**. Read; Write-1-to-clear. Reset: 0. UMI correctable error status.
3	**AbUmiGppPerr**. Read; Write-1-to-clear. Reset: 0. AB/UMI/GPP parity error status.
2	**ApuGppSerr**. Read; Write-1-to-clear. Reset: 0. APU/GPP devices SERR error status.
1	**InternalPerr**. Read; Write-1-to-clear. Reset: 0. Internal devices PERR error status.
0	**InternalSerr**. Read; Write-1-to-clear. Reset: 0. Internal devices SERR error status.

SMIx40 SciMap0

Bits	Description
31:29	Reserved.
28:24	**SciMap3**. Read-write. Reset: 0. Specifies the mapping of GEVENT3 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap0.
23:21	Reserved.
20:16	**SciMap2**. Read-write. Reset: 0. Specifies the mapping of GEVENT2 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap0.
15:13	Reserved.
12:8	**SciMap1**. Read-write. Reset: 0. Specifies the mapping of GEVENT1 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap0.

7:5	Reserved.
4:0	**SciMap0**. Read-write. Reset: 0. Specifies the mapping of GEVENT0 to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap0>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx44 SciMap1

Bits	Description
31:29	Reserved.
28:24	**SciMap7**. Read-write. Reset: 0. Specifies the mapping of GEVENT7 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap4.
23:21	Reserved.
20:16	**SciMap6**. Read-write. Reset: 0. Specifies the mapping of GEVENT6 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap4.
15:13	Reserved.
12:8	**SciMap5**. Read-write. Reset: 0. Specifies the mapping of GEVENT5 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap4.
7:5	Reserved.
4:0	**SciMap4**. Read-write. Reset: 0. Specifies the mapping of GEVENT4 to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap4>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx48 SciMap2

Bits	Description
31:29	Reserved.
28:24	**SciMap11**. Read-write. Reset: 0. Specifies the mapping of GEVENT11 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap8.
23:21	Reserved.
20:16	**SciMap10**. Read-write. Reset: 0. Specifies the mapping of GEVENT10 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap8.
15:13	Reserved.
12:8	**SciMap9**. Read-write. Reset: 0. Specifies the mapping of GEVENT9 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap8.

7:5	Reserved.
4:0	**SciMap8**. Read-write. Reset: 0. Specifies the mapping of GEVENT8 to one of the 32 bits in SMIx00 [Event_Status]. Bits: Definition 00000b: Map to the input of SMIx00 [Event_Status] bit [0]. 00001b: Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b: Map to the input of SMIx00 [Event_Status] bit [<SciMap8>]. 11111b: Map to the input of SMIx00 [Event_Status] bit [31].

SMIx4C SciMap3

Bits	Description
31:29	Reserved.
28:24	**SciMap15**. Read-write. Reset: 0. Specifies the mapping of GEVENT15 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap12.
23:21	Reserved.
20:16	**SciMap14**. Read-write. Reset: 0. Specifies the mapping of GEVENT14 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap12.
15:13	Reserved.
12:8	**SciMap13**. Read-write. Reset: 0. Specifies the mapping of GEVENT13 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap12.
7:5	Reserved.
4:0	**SciMap12**. Read-write. Reset: 0. Specifies the mapping of GEVENT12 to one of the 32 bits in SMIx00 [Event_Status]. Bits: Definition 00000b: Map to the input of SMIx00 [Event_Status] bit [0]. 00001b: Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b: Map to the input of SMIx00 [Event_Status] bit [<SciMap12>]. 11111b: Map to the input of SMIx00 [Event_Status] bit [31].

SMIx50 SciMap4

Bits	Description
31:29	Reserved.
28:24	**SciMap19**. Read-write. Reset: 0. Specifies the mapping of GEVENT19 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap16.
23:21	Reserved.
20:16	**SciMap18**. Read-write. Reset: 0. Specifies the mapping of GEVENT18 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap16.
15:13	Reserved.
12:8	**SciMap17**. Read-write. Reset: 0. Specifies the mapping of GEVENT17 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap16.

7:5	Reserved.
4:0	**SciMap16**. Read-write. Reset: 0. Specifies the mapping of GEVENT16 to one of the 32 bits in SMIx00 [Event_Status]. Bits: Definition 00000b: Map to the input of SMIx00 [Event_Status] bit [0]. 00001b: Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b: Map to the input of SMIx00 [Event_Status] bit [<SciMap16>]. 11111b: Map to the input of SMIx00 [Event_Status] bit [31].

SMIx54 SciMap5

Bits	Description
31:29	Reserved.
28:24	**SciMap23**. Read-write. Reset: 0. Specifies the mapping of GEVENT23 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap20.
23:21	Reserved.
20:16	**SciMap22**. Read-write. Reset: 0. Specifies the mapping of GEVENT22 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap20.
15:13	Reserved.
12:8	**SciMap21**. Read-write. Reset: 0. Specifies the mapping of GEVENT21 to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap20.
7:5	Reserved.
4:0	**SciMap20**. Read-write. Reset: 0. Specifies the mapping of GEVENT20 to one of the 32 bits in SMIx00 [Event_Status]. Bits: Definition 00000b: Map to the input of SMIx00 [Event_Status] bit [0]. 00001b: Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b: Map to the input of SMIx00 [Event_Status] bit [<SciMap20>]. 11111b: Map to the input of SMIx00 [Event_Status] bit [31].

SMIx58 SciMap6

Bits	Description
31:21	Reserved.
20:16	**SciMap26**. Read-write. Reset: 0. Specifies the mapping of USB_PME (device 16h) to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap24.
15:13	Reserved.
12:8	**SciMap25**. Read-write. Reset: 0. Specifies the mapping of USB_PME (device 13h) to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap24.

7:5	Reserved.
4:0	**SciMap24**. Read-write. Reset: 0. Specifies the mapping of USB_PME (device 12h) to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap24>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx5C SciMap7

Bits	Description
31:0	Reserved.

SMIx60 SciMap8

Bits	Description
31:0	Reserved.

SMIx64 SciMap9

Bits	Description
31:21	Reserved.
20:16	**SciMap38**. Read-write. Reset: 0. Specifies the mapping of SATA1_PME to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap36.
15:13	Reserved.
12:8	**SciMap37**. Read-write. Reset: 0. Specifies the mapping of SATA0_PME to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap36.
7:5	Reserved.
4:0	**SciMap36**. Read-write. Reset: 0. Specifies the mapping of HD_Audio_PME to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap36>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx68 SciMap10

Bits	Description
31:29	Reserved.
28:24	**SciMap43**. Read-write. Reset: 0. Specifies the mapping of WAKE# pin to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap40.

23:21	Reserved.
20:16	**SciMap42**. Read-write. Reset: 0. Specifies the mapping of CIR_PME to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap40.
15:13	Reserved.
12:8	**SciMap41**. Read-write. Reset: 0. Specifies the mapping of EC1_PME to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap40.
7:5	Reserved.
4:0	**SciMap40**. Read-write. Reset: 0. Specifies the mapping of EC0_PME to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap40>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx6C SciMap11

Bits	Description
31:29	Reserved.
28:24	**SciMap47**. Read-write. Reset: 0. Specifies the mapping of SMBUS0 Interrupt event to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap44.
23:21	Reserved.
20:16	**SciMap46**. Read-write. Reset: 0. Specifies the mapping of ASF slave interrupt event to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap44.
15:13	Reserved.
12:8	**SciMap45**. Read-write. Reset: 0. Specifies the mapping of ASF master interrupt event to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap44.
7:5	Reserved.
4:0	**SciMap44**. Read-write. Reset: 0. Specifies the mapping of internal FAN/THERMAL event to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap44>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx70 SciMap12

Bits	Description
31:29	Reserved.
28:24	**SciMap51**. Read-write. Reset: 0. Specifies the mapping of PWRBTN status to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap48.
23:21	Reserved.

20:16	**SciMap50**. Read-write. Reset: 0. Specifies the mapping of LLB# to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap48.
15:13	Reserved.
12:8	**SciMap49**. Read-write. Reset: 0. Specifies the mapping of internal traffic monitor to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap48.
7:5	Reserved.
4:0	**SciMap48**. Read-write. Reset: 0. Specifies the mapping of TWARN pin to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap48>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx74 SciMap13

Bits	Description
31:29	Reserved.
28:24	**SciMap55**. Read-write. Reset: 0. Specifies the mapping of RAS_event status to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap52.
23:21	Reserved.
20:16	**SciMap54**. Read-write. Reset: 0. Specifies the mapping of SCI assertion message from APU to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap52.
15:13	Reserved.
12:8	**SciMap53**. Read-write. Reset: 0. Specifies the mapping of HW assertion message from APU to one of the 32 bits in SMIx00 [Event_Status]. See: SciMap52.
7:5	Reserved.
4:0	**SciMap52**. Read-write. Reset: 0. Specifies the mapping of PROHOT# pin to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap52>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx78 SciMap14

Bits	Description
31:13	Reserved.
12:8	**SciMap57**. Read-write. Reset: 0. Specifies the mapping of XHC1 (USB3 controller 1) pin to one of the 32 bits .

7:5	Reserved.
4:0	**SciMap56**. Read-write. Reset: 0. Specifies the mapping of XHC0 (USB3 controller 0) pin to one of the 32 bits in SMIx00 [Event_Status]. Bits — Definition 00000b — Map to the input of SMIx00 [Event_Status] bit [0]. 00001b — Map to the input of SMIx00 [Event_Status] bit [1]. 11110b-00010b — Map to the input of SMIx00 [Event_Status] bit [<SciMap56>]. 11111b — Map to the input of SMIx00 [Event_Status] bit [31].

SMIx7C SciMap15

Bits	Description
31:0	Reserved.

SMIx80 SmiStatus0

Bits	Description
31:28	Reserved.
27	**ESPI_WAKE_PME_event27**. Read; Write-1-to-clear. Reset: 0. Status of eSPI Wake and Pme event.
26	**ESPI_SYS_EVT_event26**. Read; Write-1-to-clear. Reset: 0. Status of eSPI system event.
25	**UsbWakup1Event25**. Read; Write-1-to-clear. Reset: 0. Status of EHCI1Wake (bus0, Dev 20, func 0).
24	**UsbWakup0Event24**. Read; Write-1-to-clear. Reset: 0. Status of EHCI0 Wake (bus0, Dev 19, func 0).
23	**Gevent23StatusEvent23**. Read; Write-1-to-clear. Reset: 0. SMI Status of GPIO8/SerPortTX_OUT.
22	**Gevent22StatusEvent22**. Read; Write-1-to-clear. Reset: 0. SMI Status of GPIO9/SerPortRX_OUT.
21	**Gevent21StatusEvent21**. Read; Write-1-to-clear. Reset: 0. SMI Status of IR_TX0/USB_OC5_L/AGPIO13.
20	**Gevent20StatusEvent20**. Read; Write-1-to-clear. Reset: 0. SMI Status of IR_RX1/AGPIO15.
19	**Gevent19StatusEvent19**. Read; Write-1-to-clear. Reset: 0. SMI Status of SYS_RESET/AGPIO1.
18	**Gevent18StatusEvent18**. Read; Write-1-to-clear. Reset: 0. SMI Status of BLINK_L/AGPIO11/USB_OC7_L.
17	Reserved.
16	**Gevent16StatusEvent16**. Read; Write-1-to-clear. Reset: 0. SMI Status of AC_PRES/USB_OC4_L/IR_RX0/AGPIO23.
15	**Gevent15StatusEvent15**. Read; Write-1-to-clear. Reset: 0. SMI Status of TDO/USB_OC3/AGPIO24.
14	**Gevent14StatusEvent14**. Read; Write-1-to-clear. Reset: 0. SMI Status of USB_OC2/TCK/AGPIO18.
13	**Gevent13StatusEvent13**. Read; Write-1-to-clear. Reset: 0. SMI Status of USB_OC1/TDI/AGPIO17.
12	**Gevent12StatusEvent12**. Read; Write-1-to-clear. Reset: 0. SMI Status of USB_OC0/TRST_L/AGPIO16.

11	**Gevent11StatusEvent11**. Read; Write-1-to-clear. Reset: 0. SMI Status of GPIO7.
10	**Gevent10StatusEvent10**. Read; Write-1-to-clear. Reset: 0. SMI Status of AGPIO6.
9	**Gevent9StatusEvent9**. Read; Write-1-to-clear. Reset: 0. SMI Status of AGPIO68/SGPIO_CLK.
8	**Gevent8StatusEvent8**. Read; Write-1-to-clear. Reset: 0. SMI Status of WAKE_L/AGPIO2.
7	**Gevent7StatusEvent7**. Read; Write-1-to-clear. Reset: 0. SMI Status of AGPIO5/DEVSLP0.
6	**Gevent6StatusEvent6**. Read; Write-1-to-clear. Reset: 0. SMI Status of IR_TX1/AGPIO15.
5	**Gevent5StatusEvent5**. Read; Write-1-to-clear. Reset: 0. SMI Status of LPC_PD/AGPIO21.
4	**Gevent4StatusEvent4**. Read; Write-1-to-clear. Reset: 0. SMI Status of GPIO4.
3	**Gevent3StatusEvent3**. Read; Write-1-to-clear. Reset: 0. SMI Status of LPC_PME_L/AGPIO22.
2	**Gevent2StatusEvent2**. Read; Write-1-to-clear. Reset: 0. SMI Status of AGPIO3.
1	**Gevent1StatusEvent1**. Read; Write-1-to-clear. Reset: 0. SMI Status of AGPIO66.
0	**Gevent0StatusEvent0**. Read; Write-1-to-clear. Reset: 0. SMI Status of AGPIO65.

SMIx84 SmiStatus1

Bits	Description
31	**TempTsiStsEvent63**. Read; Write-1-to-clear. Reset: 0. Status of TempTsi event.
30:27	Reserved.
26	**AcDcTimerEventEvent58**. Read; Write-1-to-clear. Reset: 0. Status of AcDcTimer wake up event.
25	Reserved.
24	**Xhc0PmeEvent56**. Read; Write-1-to-clear. Reset: 0. Status of XHC0 (device 10h, function 0) PME.
23	**RasEvent55**. Read; Write-1-to-clear. Reset: 0. BIOS: See 2.17.3.9. Internal devices SERR error status.
22	**ApuSciAssertionEvent54**. Read; Write-1-to-clear. Reset: 0. Status of NB SCI request.
21	**ApuHwAssertionEvent53**. Read; Write-1-to-clear. Reset: 0. Status of APU HW assertion.
20	**ProcHotEvent52**. Read; Write-1-to-clear. Reset: 0. Status of ProcHot event.
19	**PwrButtonEvent51**. Read; Write-1-to-clear. Reset: 0. Status of PwrButton (rising edge).
18	**IllbEvent50**. Read; Write-1-to-clear. Reset: 0. Status of iLLB# assertion.
17	**TrafficMonitorIntrEvent49**. Read; Write-1-to-clear. Reset: 0. Status of FCH Traffic Monitor Interrupt.
16	**TwarnEvent48**. Read; Write-1-to-clear. Reset: 0. Status of FCH TWARN.
15	**Smbus0Event47**. Read; Write-1-to-clear. Reset: 0. Status of FCH SMBUS0 master interrupt.
14	**I2S_Waker_event46**. Read; Write-1-to-clear. Reset: 0. Status of I^2S wake event.
13	**AsfMasterIntrEvent45**. Read; Write-1-to-clear. Reset: 0. Status of FCH ASF master and slave interrupt.
12	**FanThermalGeventEvent44**. Read; Write-1-to-clear. Reset: 0. Status of FCH fan thermal.
11	**AltMmTimerStsEvent43**. Read; Write-1-to-clear. Reset: 0. Status of AltHpetTimer Alarm.
10	**CirPmeEvent42**. Read; Write-1-to-clear. Reset: 0. Status of FCH CIR PME.
9	**GpioEvent41**. Read; Write-1-to-clear. Reset: 0. Status of FCH GPIO Control interrupt.
8	**EcGevent0Event40**. Read; Write-1-to-clear. Reset: 0. Status of FCH ECG Gevent0(iEcSCi).

7	**ACP_Wake_event39**. Read; Write-1-to-clear. Reset: 0. Status of ACP wake event.
6	**SataGevent1Event38**. Read; Write-1-to-clear. Reset: 0. Status of FCH SATA Gevent1.
5	**SataGevent0Event37**. Read; Write-1-to-clear. Reset: 0. Status of FCH SATA Gevent0.
4	Reserved.
3	**FakeSts2Event33**. Read; Write-1-to-clear. Reset: 0. Status of FCH FakeSts2.
2	**FakeSts1Event33**. Read; Write-1-to-clear. Reset: 0. Status of FCH FakeSts1.
1	**FakeSts0Event33**. Read; Write-1-to-clear. Reset: 0. Status of Fake0.
0	Reserved.

SMIx88 SmiStatus2

Bits	Description
31:30	Reserved.
29	**AzFlrEvent93**. Read; Write-1-to-clear. Reset: 0. Status of Azalia FLR event.
28	**SataFlrEvent92**. Read; Write-1-to-clear. Reset: 0. Status of SATA FLR event.
27	**UsbFlrEvent91**. Read; Write-1-to-clear. Reset: 0. Status of USB FLR event.
26	**Emulate64Event90**. Read; Write-1-to-clear. Reset: 0. Status of Emulation IO Port 60h/64h.
25	**ThermaltripEvent89**. Read; Write-1-to-clear. Reset: 0. Status of ThermalTrip event.
24:22	Reserved.
21	**GppSerr0Event84**. Read; Write-1-to-clear. Reset: 0. SERR error from FCH GPP device 15h, function 0~3.
20	**PciSerrEvent84**. Read; Write-1-to-clear. Reset: 0. Status of Serr assertion on PCI bus.
19	**ProtHotEvent83**. Read; Write-1-to-clear. Reset: 0. Status of ProtHot event.
18	**VbatLowEvent82**. Read; Write-1-to-clear. Reset: 0. Status of VBAT low.
17	**IntruderAlertSts_event81**. Read; Write-1-to-clear. Reset: 0. Status of Intruder Alert event.
16	**XhcErrEvent80**. Read; Write-1-to-clear. Reset: 0. Status of XHC error.
15	**EcSmi0Event79**. Read; Write-1-to-clear. Reset: 0. Status of Integrated Micro-Controller (IMC) SMI request.
14	**Smbus0IntrEvent78**. Read; Write-1-to-clear. Reset: 0. Status of SMBUS0 interrupt request.
13	**SerialIrqSmiEvent77**. Read; Write-1-to-clear. Reset: 0. Status of SMI request from Serial IRQ.
12	**UsbSmiEvent76**. Read; Write-1-to-clear. Reset: 0. Status of USB SMI request.
11	**SmiCmdportEvent75**. Read; Write-1-to-clear. Reset: 0. Status of Writing SMI Command Port.
10	**PwrbtnEvent74**. Read; Write-1-to-clear. Reset: 0. Status of Power Button being pressed.
9	**BiosRlsEvent73**. Read; Write-1-to-clear. Reset: 0. Status of BIOS_RLS. See PMx74[BiosRls].
8	**GblRlsEvent72**. Read; Write-1-to-clear. Reset: 0. Status of GBL event. See AcpiPm1CntBlkx00[GblRls].
7:4	Reserved.
3	**SataAhciSmiEvent67**. Read; Write-1-to-clear. Reset: 0. Status of SATA AHCI SMI request.
2	**AL2H_ACPI_Assertion_Event67**. Read; Write-1-to-clear. Reset: 0. Status of AL2H ACPI Assertion.

1	**SlpTypeEvent65**. Read; Write-1-to-clear. Reset: 0. Status of writing AcpiPm1CntBlkx00[SlpTyp].
0	**KbRstEvent64**. Read; Write-1-to-clear. Reset: 0. Status of KeyBoard Reset event.

SMIx8C SmiStatus3

Bits	Description
31:0	Reserved.

SMIx90 SmiStatus4

Bits	Description
31:29	Reserved.
28	**CfgTrapping0Event156**. Read; Write-1-to-clear. Reset: 0. Status of PCI configuration cycle Trapping0 SMI request.
27:25	Reserved.
24	**MemTrapping0Event152**. Read; Write-1-to-clear. Reset: 0. Status of memory Trapping0 SMI request.
23:21	Reserved.
20	**IoTrapping0Event148**. Read; Write-1-to-clear. Reset: 0. Status of IO Trapping0 SMI request.
19:18	Reserved.
17	**SoftResetEvent145**. Read; Write-1-to-clear. Reset: 0. Status of soft reset request. The soft request can be: • CF9h PCI reset. • ASF remote reset. • sync flood reset. • KBRST. • watchdog timer reset.
16	**AbSmiTrapEvent144**. Read; Write-1-to-clear. Reset: 0. Status of AB SMI trapping request.
15	**LongTimerEvent143**. Read; Write-1-to-clear. Reset: 0. Status of Long Timer SMI request.
14	**ShortTimerEvent142**. Read; Write-1-to-clear. Reset: 0. Status of Short Timer SMI request.
13	Reserved.
12	**Fake2StsEvent140**. Read; Write-1-to-clear. Reset: 0. Status of Fake2.
11	**Fake1StsEvent139**. Read; Write-1-to-clear. Reset: 0. Status of Fake1.
10	**Fake0StsEvent138**. Read; Write-1-to-clear. Reset: 0. Status of Fake0.
9:6	Reserved.
5	**Fanin0stsEvent133**. Read; Write-1-to-clear. Reset: 0. Status of FanIn0 event.
4:0	Reserved.

SMIx94 SmiPointer

This register is meant as a faster mechanism to locate the SMI source. BIOS can examine this register to find out the SMI source instead of reading SMIx80 [SmiStatus0] through SMIx90 [SmiStatus4] individually.

Bits	Description
15:6	Reserved.
5	**SmiStatusSource4**. Read-only. Reset: 0. Indicates whether the SMI source is from SMIx90 [SmiStatus4] if the corresponding SMI enable is selected.
4	**SmiStatusSource3**. Read-only. Reset: 0. Indicates whether the SMI source is from SMIx8C [SmiStatus3] if the corresponding SMI enable is selected.
3	**SmiStatusSource2**. Read-only. Reset: 0. Indicates whether the SMI source is from SMIx88 [SmiStatus2] if the corresponding SMI enable is selected.
2	**SmiStatusSource1**. Read-only. Reset: 0. Indicates whether the SMI source is from SMIx84 [SmiStatus1] if the corresponding SMI enable is selected.
1	**SmiStatusSource0**. Read-only. Reset: 0. Indicates whether the SMI source is from SMIx80 [SmiStatus0] if the corresponding SMI enable is selected.
0	**SmiSciSource**. Read-only. Reset: 0. Indicates whether the SMI source is from SMIx10 [SmiSciStatus].

SMIx96 SmiTimer

There are two SMI timers: the short timer runs at 1 us unit time and the long timer runs at 1 ms unit time. This register is actually made up of two sets of registers depending on the setting of SMIx98[SmiTimerSel]. If SMIx98[SmiTimerSel] == 0, then SMIx96 is for the short timer. If SMIx98[SmiTimerSel] == 1, then SMIx96 is for the long timer. The default setting selects this register as “SmiShortTimer”; software needs to set SMIx98[SmiTimerSel] = 1 to select this register as “SmiLongTimer”.

Bits	Description
15	**TimerEn**. Read-write. Reset: 0. 1=Enable the SMI short Timer or long timer, which is selected by SMIx98[SmiTimerSel]. 0=Disable.
14:0	**SmiTimerCount**. Read-write. Reset: 0. Actual timer duration = TimerTime + 1 unit.

SMIx98 SmiTrig0

Bits[23:0] defines the trigger mode for 24 GEVENTS in SMIx80[23:0]. Note these are different from SMIx08 [SciTrig].

Bits	Description
31	**SmiEnB**. Read-write. Reset: 1. 0=Enable SMI function. 1=Disable.
30	Reserved.
29	**SmiTimerSel**. Read-write. Reset: 0. 0=Select SMIx96 [SmiTimer] to be SmiShortTimer register. 1=Select SMIx96 [SmiTimer] to be SmiLongTimer register.
28	**Eos**. Read-write. Reset: 0. 1=Allow SMI to be sent out to CPU. 0=SMI is blocked.
27	**FakeSts2**. Read-write. Reset: 1. Program the value to emulate an SMI input event.
26	**FakeSts1**. Read-write. Reset: 1. Program the value to emulate an SMI input event.

25	**FakeSts0**. Read-write. Reset: 1. Program the value to emulate an SMI input event.
24	**TrappingIrqOnPic**. Read-write. Reset: 1. 0=SMI is generated when trapping IRQ0-15 of IOAPIC. 1=SMI is generated when trapping IRQ0-15 of PIC.
23	**SmiTrig23**. Read-write. Reset: 1. 0=Active low. 1=Active high.
22	**SmiTrig22**. Read-write. Reset: 1. 0=Active low. 1=Active high.
21	**SmiTrig21**. Read-write. Reset: 1. 0=Active low. 1=Active high.
20	**SmiTrig20**. Read-write. Reset: 1. 0=Active low. 1=Active high.
19	**SmiTrig19**. Read-write. Reset: 1. 0=Active low. 1=Active high.
18	**SmiTrig18**. Read-write. Reset: 1. 0=Active low. 1=Active high.
17	**SmiTrig17**. Read-write. Reset: 1. 0=Active low. 1=Active high.
16	**SmiTrig16**. Read-write. Reset: 1. 0=Active low. 1=Active high.
15	**SmiTrig15**. Read-write. Reset: 1. 0=Active low. 1=Active high.
14	**SmiTrig14**. Read-write. Reset: 1. 0=Active low. 1=Active high.
13	**SmiTrig13**. Read-write. Reset: 1. 0=Active low. 1=Active high.
12	**SmiTrig12**. Read-write. Reset: 1. 0=Active low. 1=Active high.
11	**SmiTrig11**. Read-write. Reset: 1. 0=Active low. 1=Active high.
10	**SmiTrig10**. Read-write. Reset: 1. 0=Active low. 1=Active high.
9	**SmiTrig9**. Read-write. Reset: 1. 0=Active low. 1=Active high.
8	**SmiTrig8**. Read-write. Reset: 1. 0=Active low. 1=Active high.
7	**SmiTrig7**. Read-write. Reset: 1. 0=Active low. 1=Active high.
6	**SmiTrig6**. Read-write. Reset: 1. 0=Active low. 1=Active high.
5	**SmiTrig5**. Read-write. Reset: 1. 0=Active low. 1=Active high.
4	**SmiTrig4**. Read-write. Reset: 1. 0=Active low. 1=Active high.
3	**SmiTrig3**. Read-write. Reset: 1. 0=Active low. 1=Active high.
2	**SmiTrig2**. Read-write. Reset: 1. 0=Active low. 1=Active high.
1	**SmiTrig1**. Read-write. Reset: 1. 0=Active low. 1=Active high.
0	**SmiTrig0**. Read-write. Reset: 1. 0=Active low. 1=Active high.

SMIx9C SmiTrig1

Bits[23:0] defines the trigger mode for 24 IRQs.

Bits	Description
31:24	Reserved.
23	**SmiIrq23Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
22	**SmiIrq22Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
21	**SmiIrq21Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
20	**SmiIrq20Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
19	**SmiIrq19Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
18	**SmiIrq18Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.

17	**SmiIrq17Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
16	**SmiIrq16Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
15	**SmiIrq15Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
14	**SmiIrq14Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
13	**SmiIrq13Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
12	**SmiIrq12Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
11	**SmiIrq11Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
10	**SmiIrq10Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
9	**SmiIrq9Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
8	**SmiIrq8Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
7	**SmiIrq7Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
6	**SmiIrq6Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
5	**SmiIrq5Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
4	**SmiIrq4Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
3	**SmiIrq3Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
2	**SmiIrq2Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
1	**SmiIrq1Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.
0	**SmiIrq0Trig**. Read-write. Reset: 0. 0=Active low. 1=Active high.

SMIxA0 SmiControl0

This register specifies the control mechanism for SMI sources in SMIx80[15:0].

Bits	Description
31:30	**Smicontrol15**. Read-write. Reset: 0. Control for GEVENT15. See: Smicontrol0.
29:28	**Smicontrol14**. Read-write. Reset: 0. Control for GEVENT14. See: Smicontrol0.
27:26	**Smicontrol13**. Read-write. Reset: 0. Control for GEVENT13. See: Smicontrol0.
25:24	**Smicontrol12**. Read-write. Reset: 0. Control for GEVENT12. See: Smicontrol0.
23:22	**Smicontrol11**. Read-write. Reset: 0. Control for GEVENT11. See: Smicontrol0.
21:20	**Smicontrol10**. Read-write. Reset: 0. Control for GEVENT10. See: Smicontrol0.
19:18	**Smicontrol9**. Read-write. Reset: 0. Control for GEVENT9. See: Smicontrol0.
17:16	**Smicontrol8**. Read-write. Reset: 0. Control for GEVENT8. See: Smicontrol0.
15:14	**Smicontrol7**. Read-write. Reset: 0. Control for GEVENT7. See: Smicontrol0.
13:12	**Smicontrol6**. Read-write. Reset: 0. Control for GEVENT6. See: Smicontrol0.
11:10	**Smicontrol5**. Read-write. Reset: 0. Control for GEVENT5. See: Smicontrol0.
9:8	**Smicontrol4**. Read-write. Reset: 0. Control for GEVENT4. See: Smicontrol0.
7:6	**Smicontrol3**. Read-write. Reset: 0. Control for GEVENT3. See: Smicontrol0.
5:4	**Smicontrol2**. Read-write. Reset: 0. Control for GEVENT2. See: Smicontrol0.

3:2	**Smicontrol1**. Read-write. Reset: 0. Control for GEVENT1. See: Smicontrol0.
1:0	**Smicontrol0**. Read-write. Reset: 0. Control for GEVENT0. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13

SMIxA4 SmiControl1

This register specifies the control mechanism for SMI sources in SMIx80[31:16].

Bits	Description
31:30	**Smicontrol31**. Read-write. Reset: 0. Control for GPP PME(function 3, Device 15h).
29:28	**Smicontrol30**. Read-write. Reset: 0. Control for GPP PME(function 2, Device 15h).
27:26	**Smicontrol29**. Read-write. Reset: 0. Control for GPP PME(function 1, Device 15h).
25:24	**Smicontrol28**. Read-write. Reset: 0. Control for any GPP PME(Device 15h).
23:18	Reserved.
17:16	**Smicontrol24**. Read-write. Reset: 0. Control for USB_PME (Device 12h). See: Smicontrol16.
15:14	**Smicontrol23**. Read-write. Reset: 0. Control for GEVENT23. See: Smicontrol16.
13:12	**Smicontrol22**. Read-write. Reset: 0. Control for GEVENT22. See: Smicontrol16.
11:10	**Smicontrol21**. Read-write. Reset: 0. Control for GEVENT21. See: Smicontrol16.
9:8	**Smicontrol20**. Read-write. Reset: 0. Control for GEVENT20. See: Smicontrol16.
7:6	**Smicontrol19**. Read-write. Reset: 0. Control for GEVENT19. See: Smicontrol16.
5:4	**Smicontrol18**. Read-write. Reset: 0. Control for GEVENT18. See: Smicontrol16.
3:2	**Smicontrol17**. Read-write. Reset: 0. Control for GEVENT17. See: Smicontrol16.
1:0	**Smicontrol16**. Read-write. Reset: 0. Control for GEVENT16. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13

SMIxA8 SmiControl2

This register specifies the control mechanism for SMI sources in SMIx84[15:0].

Bits	Description
31:30	**Smicontrol47**. Read-write. Reset: 0. Control for SMBUS0 interrupt. See: Smicontrol36.
29:28	**Smicontrol46**. Read-write. Reset: 0. Control for ASF Slave interrupt. See: Smicontrol36.
27:26	**Smicontrol45**. Read-write. Reset: 0. Control for ASF Master interrupt. See: Smicontrol36.
25:24	**Smicontrol44**. Read-write. Reset: 0. Control for fan thermal GEvent. See: Smicontrol36.
23:22	**SmiControl43**. Read-write. Reset: 0. Control for ALTHPET_TimerSts. See: Smicontrol36.
21:20	**Smicontrol42**. Read-write. Reset: 0. Control for CIR_PME. See: Smicontrol36.

19:18	**SmiControl41**. Read-write. Reset: 0. Control for GPIO interrupt. See: Smicontrol36.
17:16	**Smicontrol40**. Read-write. Reset: 0. Control for IMC Gevent0. See: Smicontrol36.
15:14	Reserved.
13:12	**Smicontrol38**. Read-write. Reset: 0. Control for SATA Gevent1. See: Smicontrol36.
11:10	**Smicontrol37**. Read-write. Reset: 0. Control for SATA Gevent0. See: Smicontrol36.
9:8	**Smicontrol36**. Read-write. Reset: 0. Control for PME from HD Audio. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13
7:6	**Smicontrol35**. Read-write. Reset: 0. Control for FakeSts2.
5:4	**Smicontrol34**. Read-write. Reset: 0. Control for FakeSts1.
3:2	**Smicontrol33**. Read-write. Reset: 0. Control for FakeSts0.
1:0	**Smicontrol32**. Read-write. Reset: 0. Control for GPP HP (function 0-3, Device 15h).

SMIxAC SmiControl3

This register specifies the control mechanism for SMI sources in SMIx84[31:16].

Bits	Description
31:30	**Smicontrol63**. Read-write. Reset: 0. Control for TempTsi event. See: Smicontrol48.
29:22	Reserved.
21:20	**Smicontrol58**. Read-write. Reset: 0. Control for AcDcTimer wake up event (Wake Device in ACPI4.0). See: Smicontrol48.
19:18	Reserved.
17:16	**Smicontrol56**. Read-write. Reset: 0. Control for XHC0 (device 10h, function 0) PME. See: Smicontrol48.
15:14	**Smicontrol55**. Read-write. Reset: 0. Control for internal devices SERR error status. See: Smicontrol48.
13:12	**Smicontrol54**. Read-write. Reset: 0. Control for APU SCI request. See: Smicontrol48.
11:10	**Smicontrol53**. Read-write. Reset: 0. Control for APU HW assertion. See: Smicontrol48.
9:8	**Smicontrol52**. Read-write. Reset: 0. Control for ProcHot event. See: Smicontrol48.
7:6	**Smicontrol51**. Read-write. Reset: 0. Control for Power button event. See: Smicontrol48.
5:4	**Smicontrol50**. Read-write. Reset: 0. Control for iLLB#. See: Smicontrol48.
3:2	**Smicontrol49**. Read-write. Reset: 0. Control for internal Traffic monitor interrupt. See: Smicontrol48.
1:0	**Smicontrol48**. Read-write. Reset: 0. Control for TWARN#. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13

SMIxB0 SmiControl4

This register specifies the control mechanism for SMI sources in SMIx88[15:0].

Bits	Description
31:30	**Smicontrol79**. Read-write. Reset: 0. Control for IMC SMI request0. See: Smicontrol65.
29:28	**Smicontrol78**. Read-write. Reset: 0. Control for SMBUS0 interrupt. See: Smicontrol65.
27:26	**Smicontrol77**. Read-write. Reset: 0. Control for SMI request form serial IRQ. See: Smicontrol65.
25:24	**Smicontrol76**. Read-write. Reset: 0. Control for USB SMI request. See: Smicontrol65.
23:22	**Smicontrol75**. Read-write. Reset: 0. Control for writing SMI command port at SmiCmdBlkx00. See: Smicontrol65.
21:20	**Smicontrol74**. Read-write. Reset: 0. Control for power button being pressed. See: Smicontrol65.
19:18	**Smicontrol73**. Read-write. Reset: 0. Control for writing PMx74[BiosRls]. See: Smicontrol65.
17:16	**Smicontrol72**. Read-write. Reset: 0. Control for writing AcpiPm1CntBlkx00[GblRls]. See: Smicontrol65.
15:8	Reserved.
7:6	**Smicontrol67**. Read-write. Reset: 0. Control for SATA AHCI event. See: Smicontrol65.
5:4	**Smicontrol66**. Read-write. Reset: 0. Control for iAL2H ACPI Assertion. See: Smicontrol65.
3:2	**Smicontrol65**. Read-write. Reset: 0. Control for writing AcpiPm1CntBlkx00[SlpTyp] to put the system in S-state. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13
1:0	**Smicontrol64**. Read-write. Reset: 0. Control for Keyboard Reset event. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13

SMIxB4 SmiControl5

This register specifies the control mechanism for SMI sources in SMIx88[31:16].

Bits	Description
31:28	Reserved.
27:26	**Smicontrol93**. Read-write. Reset: 0. Control for HD audio FLR. See: Smicontrol80.
25:24	**Smicontrol92**. Read-write. Reset: 0. Control for SATA FLR. See: Smicontrol80.
23:22	**Smicontrol91**. Read-write. Reset: 0. Control for USB FLR. See: Smicontrol80.
21:20	**Smicontrol90**. Read-write. Reset: 0. Control for Emulation64. See: Smicontrol80.
19:18	**Smicontrol89**. Read-write. Reset: 0. Control for ThermalTrip# assertion. See: Smicontrol80.
17:12	Reserved.

11:10	**Smicontrol85**. Read-write. Reset: 0. Control for SB any GPP Serr#. See: Smicontrol80.
9:8	**Smicontrol84**. Read-write. Reset: 0. Control for SERR#. See: Smicontrol80.
7:6	**Smicontrol83**. Read-write. Reset: 0. Control for ProcHot. See: Smicontrol80.
5:4	**Smicontrol82**. Read-write. Reset: 0. Control for VBAT low. See: Smicontrol80.
3:2	**Smicontrol81**. Read-write. Reset: 0. Control for Intruder event. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13
1:0	**Smicontrol80**. Read-write. Reset: 0. Control for IMC SMI request1. Bits Definition 00b Disable 01b SMI 10b NMI 11b IRQ13

SMIxB8 SmiControl6

This register specifies the control mechanism for SMI sources in SMIx8C[15:0].

Bits	Description
31:0	Reserved.

SMIxBC SmiControl7

This register specifies the control mechanism for SMI sources in SMIx8C[31:16].

Bits	Description
31:0	Reserved.

SMIxC0 SmiControl8

This register specifies the control mechanism for SMI sources in SMIx90[15:0].

Bits	Description
31:30	**Smicontrol143**. Read-write. Reset: 0. Control for Long timer. See: Smicontrol133.
29:28	**Smicontrol142**. Read-write. Reset: 0. Control for Short timer. See: Smicontrol133.
27:26	Reserved.
25:24	**Smicontrol140**. Read-write. Reset: 0. Control for SMIx98[FakeSts2] generated interrupts. See: Smicontrol133.
23:22	**Smicontrol139**. Read-write. Reset: 0. Control for SMIx98[FakeSts1] generated interrupts. See: Smicontrol133.
21:20	**Smicontrol138**. Read-write. Reset: 0. Control for SMIx98[FakeSts0] generated interrupts. See: Smicontrol133.
19:12	Reserved.

11:10	**Smicontrol133**. Read-write. Reset: 0. Control for Fan0 Tach too slow event. Bits: Definition 00b: Disable 01b: SMI 10b: NMI 11b: IRQ 13
9:0	Reserved.

SMIxC4 SmiControl9

This register specifies the control mechanism for SMI sources in SMIx90[31:16].

Bits	Description
31:26	Reserved.
25:24	**Smicontrol156**. Read-write. Reset: 0. Control for configuration cycle trapping 0. See: Smicontrol144.
23:18	Reserved.
17:16	**Smicontrol152**. Read-write. Reset: 0. Control for memory trapping 0. See: Smicontrol144.
15:10	Reserved.
9:8	**Smicontrol148**. Read-write. Reset: 0. Control for IO trapping 0. See: Smicontrol144.
7:4	Reserved.
3:2	**Smicontrol145**. Read-write. Reset: 0. Control for soft reset request. See: Smicontrol144.
1:0	**Smicontrol144**. Read-write. Reset: 0. Control for AB SMI trapping request. Bits: Definition 00b: Disable 01b: SMI 10b: NMI 11b: IRQ13

SMIxC8 IoTrapping0

Bits	Description
15:0	**IoTrapping0**. Read-write. Reset: 0. Specifies the IO address which causes SMI event.

SMIxD0 MemTrapping0

Bits	Description
31:2	**MemTrapping**. Read-write. Reset: 0. Specifies the 30-bit memory address which causes SMI event. The lowest 2 bits are ignored.
1	**MemRdOvrEn**. Read-write. Reset: 0. 1=Force read data to be replaced by SMIxD4 [MemRdOvrData0].
0	**MemTrappingRw**. Read-write. Reset: 0. 0=Trap on read access at the address specified by [MemTrapping]. 1=Trap on write access.

SMIxD4 MemRdOvrData0

Bits	Description
31:0	**MemRdOvrData**. Read-write. Reset: 0. This 32-bit data is used as the return data when the memory read trapping is enabled through SMIxD0[MemRdOvrEn].

SMIxF0 CfgTrapping0

Bits	Description
31:2	**CfgTrapping**. Read-write. Reset: 0. Specifies the Address[31:2] value during configure cycle which causes SMI event.
1	**IoTrappingRw0**. Read-write. Reset: 0. 0=Trap on IO read access on the address specified in SMIxC8 [IoTrapping0]. 1=Trap on IO write access on the address specified in SMIxC8 [IoTrapping0].
0	**CfgTrappingRw**. Read-write. Reset: 0. 0=Trap on read access at the address specified in [CfgTrapping]. 1=Trap on write access.

3.26.16 Watchdog Timer (WDT) Registers

Watchdog timer registers are accessed through two methods:
- Memory access to Watchdog Timer memory address range from FEB0_0000h to FEB0_000Fh. Program PMx00[WatchdogTmrEn] = 1 to enable Watchdog Timer decoding.
- Memory mapped access through the AcpiMmio region. The Watchdog Timer registers start from FED8_0000h+B00h. See PMx04[MmioEn].

WDTx00 WatchdogControl

Bits	Description
31:8	Reserved.
7	**WatchdogTrigger**. IF (!WatchdogDisable && WatchdogRunStopB) THEN RAZ; Write-1-only. ELSE RAZ. ENDIF. Reset: 0. Writing 1 to this bit triggers the watchdog to start a new count interval, counting down from the value that was last written to WDTx04.
6:4	Reserved.
3	**WatchdogDisable**. Read-only. Reset: 1. This bit reflects the state of the watchdog timer hardware. 0=Enable. 1=Disable.
2	**WatchdogAction**. Read-write. Reset: 0. This bit determines the action to be taken when the watchdog timer expires. 0=System reset. 1=System power off. The bit is only valid when the watchdog is enabled.
1	**WatchdogFired**. Read; Write-1-to-clear. Cold reset: 0. 1=The watchdog timer has expired and caused the current restart. The bit is cleared by a power cycle or by the operating system and it must remain cleared for any restart that is not caused by the watchdog timer firing. The bit is only valid when the watchdog is enabled.
0	**WatchdogRunStopB**. Read-write. Reset: 0. 1=Watchdog is in the running state. 0=Watchdog is in the stopped state. This bit is used to control or indicate whether the watchdog is in the running or stopped states. If the watchdog is in the stopped state and a 1 is written to this bit, the watchdog moves to the running state, but a count interval is not started until a 1 is written to WatchdogTrigger. If the watchdog is in the running state, writing 1 to this bit has no effect. The bit is only valid when the watchdog is enabled.

WDTx04 WatchdogCount

Bits	Description
31:16	Reserved.
15:0	**WatchdogCount**. Read-write. Reset: X. Writing this field specifies the countdown time for the counter.

3.26.17 Wake Alarm Device (AcDcTimer) Registers

The AC/DC timer registers are used to control the wake alarm device. They are accessed through the AcpiMmio region. The AC DC timer registers range from FED8_0000h+1D00h to FED8_0000h+1DFFh. See PMx04[MmioEn].

AcDcTimerx00 AcTimerValue

Bits	Description
31:0	**AcTimerValue**. Read-write; Updated-by-hardware. Reset: FFFF_FFFFh. Writing the register with a value other than FFFF_FFFFh starts the AC timer. Reading this register returns the current value of the AC timer. When the AC or DC timer generates a wake up event, this register is reset to FFFF_FFFFh by hardware. Bits — Definition 0000_0000h — The AC timer wakes up the system instantly. FFFF_FFFEh-0000_0001h — The value indicates the number of seconds between the time when the AC timer is programmed and the time when it expires. FFFF_FFFFh — Disable the AC timer.

AcDcTimerx04 AcExpiredTimerPolicy

Bits	Description
31:0	**AcExpiredTimerPolicy**. Read-write; Updated-by-hardware. Reset: FFFF_FFFFh. When the AC timer expires, the wake signal is asserted if the current power source is AC; the wake signal is not asserted if the current power source is DC. If the power source is switched from DC to AC after the AC timer expired, we wait for the number of seconds defined in this register and then wake up the system. When the AC or DC timer generates a wake up event, this register is reset to FFFF_FFFFh by hardware. Bits — Definition 0000_0000h — The expired AC timer wakes up the system instantly once the power source is switched to AC. FFFF_FFFEh-0000_0001h — The value indicates the number of seconds between the time when the power is switched to AC and the time when it generates the wake-up event. FFFF_FFFFh — Disable the AC expired timer policy.

AcDcTimerx08 AcTimerStatus

Bits	Description
31:2	Reserved.
1	**AcTimerWakeup**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Wake up was caused by AC timer expiration. 0=Wake-up was not caused by AC timer expiration.
0	**AcTimerExpired**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=AC timer expired. 0=AC timer has not expired.

AcDcTimerx10 DcTimerValue

Bits	Description
31:0	**DcTimerValue**. Read-write; Updated-by-hardware. Reset: FFFF_FFFFh. Writing the register with a value other than FFFF_FFFFh starts the DC timer. Reading this register returns the current value of the DC timer. When the AC or DC Timer generates a wake up event, this register is reset to FFFF_FFFFh by hardware. Bits — Definition 0000_0000h — The DC timer wakes up the system instantly. FFFF_FFFEh-0000_0001h — The value indicates the number of seconds between the time when the DC timer is programmed and the time when it expires. FFFF_FFFFh — Disable the DC timer.

AcDcTimerx14 DcExpiredTimerPolicy

Bits	Description
31:0	**DcExpiredTimerPolicy**. Read-write; Updated-by-hardware. Reset: FFFF_FFFFh. When the DC timer expires, the wake signal is asserted if the current power source is DC; the wake signal is not asserted if the current power source is AC. If the power source is switched from AC to DC after the DC timer expired, we wait for the number of seconds defined in this register and then wake up the system. When the AC or DC timer generates a wake up event, this register is reset to FFFF_FFFFh by hardware. Bits — Definition 0000_0000h — The expired DC timer wakes up the system instantly once the power source is switched to DC. FFFF_FFFEh-0000_0001h — The value indicates the number of seconds between the time when the power is switched to DC and the time when it generates the wake-up event. FFFF_FFFFh — Disable the DC expired timer policy.

AcDcTimerx18 DcTimerStatus

Bits	Description
31:2	Reserved.
1	**DcTimerWakeup**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=Wake-up was caused by DC timer expiration. 0=Wake-up was not caused by DC timer expiration.
0	**DcTimerExpired**. Read; Set-by-hardware; Write-1-to-clear. Reset: 0. 1=DC timer expired. 0=DC timer has not expired.

AcDcTimerx20 AcDcTimerCtrl

Bits	Description
31:13	Reserved.
12	**No_event_in_G0**. Read-write. Reset: 1. Debug purpose.

11:10	**Sel_wake_rst**. Read-write. Reset: 0. Debug purpose.
9	**DcTimerEventEn**. Read-write. Reset: 0. 1=Enable DC Timer to wake up system. 0=Disable.
8	**AcTimerEventEn**. Read-write. Reset: 0. 1=Enable AC Timer to wake up system. 0=Disable.
7:1	Reserved.
0	**Busy**. Read-only; Updated-by-hardware. Reset: 0h. Right after AcDcTimerx00 [AcTimerValue] or AcDcTimerx10 [DcTimerValue] is programed, the hardware sets this bit to 1. The hardware clears this bit once the programming is done and the corresponding timer is started properly. Before the software writes to AcDcTimerx00 [AcTimerValue] or AcDcTimerx10 [DcTimerValue], it has to read this bit and make sure it is 0. Otherwise, the hardware ignores the programming action from software. For the registers other than AcDcTimerx00 [AcTimerValue] or AcDcTimerx10 [DcTimerValue], there is no such limitation.

3.26.18 A-Link to AHB Bridge (AL2AHB) Configuration Registers

The AL2AHB registers are memory-mapped. They range from FEDC_0000h to FEDC_0FFFh.

AL2AHBx00 Hard Address Low

Bits	Description
31:16	**HardAddrLow**. Read-write. Reset: FEDCh. Defines bits[31:16] of the slave base address register above which all AMBA fabric and ACPI registers are accessed.
15:0	Reserved.

AL2AHBx04 Hard Address High

Bits	Description
31:0	**HardAddrHigh**. Read-write. Reset: 0000_0000h. Defines bits[63:32] of the slave base address register above which all AMBA fabric and ACPI registers are accessed.

AL2AHBx10 Control

Bits	Description
31:13	**AL2AHB_Control_31_13**. Read-write. Reset: 0. Control specific to the AL2AHB bridge.
12	**DSM_STALL_EN**. Read-write. Reset: 0. 1=Enable external DSM stall mechanism.
11:10	**AL2AHB_Control_11_10**. Read-write. Reset: 0. Control specific to the AL2AHB bridge.
9:2	**CLK_GATE_TIME**. Read-write. Reset: 4. Delay cycle count to insert after idle condition before gating ALCLK.
1	**ClkGateEn**. Read-write. Reset: 0. 1=Enable internal clock gating for ALCLK domain logic.
0	**CdcbufenclkEn**. Read-write. Reset: 1. 1=Enable CDCBUFENCLK.

AL2AHBx14 Status

Bits	Description
31:0	**Al2AhbStatus**. Read-write. Reset: 0000_0000h.

AL2AHBx18 Mask

Bits	Description
31:0	**Al2AhbMask**. Read-write. Reset: 0000_0000h.

AL2AHBx20 HClk Hard Address Low

Bits	Description
31:16	**HClkHardAddrLow**. Read-write. Reset: FEDCh. Defines bits[31:16] of the slave base address register above which all AMBA fabric and ACPI registers are to be accessed.
15:0	Reserved.

AL2AHBx24 HClk Hard Address High

Bits	Description
31:0	**HClkHardAddrHigh**. Read-write. Reset: 0000_0000h. Defines bits[63:32] of the slave base address register above which all AMBA fabric and ACPI registers are to be accessed.

AL2AHBx30 HClk Control

Bits	Description
31:2	**HClkAl2AhbControl**. Read-write. Reset: 0. Control specific to the AL2AHB device.
1	**HClkGateEn**. Read-write. Reset: 0. 1=Enable internal clock gating for HCLK domain logic.
0	**HClkCdcbufenclkEn**. Read-write. Reset: 1. 1=Enable CDCBUFENCLK.

AL2AHBx34 HClk Status

Bits	Description
31:0	**HClkAl2AhbStatus**. Read-write. Reset: 0000_0000h.

AL2AHBx38 HClk Mask

Bits	Description
31:0	**HClkAl2AhbMask**. Read-write. Reset: 0000_0000h.

AL2AHBx40 AMBA Control 1

Bits	Description
31:6	Reserved.
5	**RegsCplStopAPBTimeOut**. Read-write. Reset: 0. 1=Disable APB timeout.
4	**RegsCplEnRddata**. Read-write. Reset: 0. 1=Enable RdData stall.
3	**RegsCplEnARaddr**. Read-write. Reset: 0. 1=Enable ARAddr stall.
2	**RegsCplEnAWaddr**. Read-write. Reset: 0. 1=Enable AWAddr stall.
1	**RegsCplStopClkEnable**. Read-write. Reset: 0. 1=Disable clock gating.
0	Reserved.

AL2AHBx44 AMBA Control 2

Reset: 0000_0000h.

Bits	Description
31:24	**RegsH2X1WrDatFifoDepth**. Read-write. Specifies the upstream AHB to AXI #1 write data FIFO depth.
23	Reserved.
22	**RegsH2X1WaitBResp**. Read-write. 1=For upstream AHB to AXI #1, along with RegsH2XPciOrdering, reads wait until all pending write responses have been received before sending onto AXI Read Address Channel.
21	**RegsH2X1PciOrdering**. Read-write. 1=For upstream AHB to AXI #1, the read command FIFO is not read until all writes issued before the read have been transferred on the Write Address Channel.
20	**RegsH2X1NoAwCache**. Read-write. Upstream AHB to AXI #1 zero out AWCACHE.
19	**RegsH2X1Wrap2Incr**. Read-write. Upstream AHB to AXI #1 convert all wrap burst to incr burst.
18	**RegsH2X1AlignCacheline**. Read-write. Upstream AHB to AXI #1 align to cache line boundary (64-byte).
17	**RegsH2X1FullWrBurst**. Read-write. Upstream AHB to AXI #1 wait for write burst data before issuing AXI write.
16	**RegsH2X1WriteCombineOff**. Read-write. 1=Disable upstream AHB to AXI #1 write combine.
15:8	**RegsH2x0WrDatFifoDepth**. Read-write. Upstream AHB to AXI #0 write data FIFO depth.
7:5	Reserved.
4	**RegsH2X0NoAwCache**. Read-write. Upstream AHB to AXI #0 zero out AWCACHE.
3	**RegsH2X0Wrap2Incr**. Read-write. Upstream AHB to AXI #0 convert all wrap burst to incr burst.
2	**RegsH2X0AlignCacheline**. Read-write. Upstream AHB to AXI #0 align to cache line boundary (64-byte).
1	**RegsH2X0FullWrBurst**. Read-write. Upstream AHB to AXI #0 wait for write burst data before issuing AXI write.
0	**RegsH2X0WriteCombineOff**. Read-write. 1=Disable upstream AHB to AXI #0 write combine.

AL2AHBx50 AMBA Status 1

Reset: 0000_0000h.

Bits	Description
31:18	Reserved.
17	**X2XRegsNonModErr**. Read-write. AXI to AXI.
16	**X2XRegsBufRespErr**. Read-write. AXI to AXI bufferable response with error.
15:0	Reserved.

AL2AHBx54 AMBA Status 2

Reset: 0000_0000h.

Bits	Description
31:30	Reserved.
29	**I2C1RegsH2PPostingErr**. Read-write. I^2C #1 AHB to APB write posting with error.
28	**I2C0RegsH2PPostingErr**. Read-write. I^2C #0 AHB to APB write posting with error.
27:26	Reserved.
25	**Uart1RegsH2PPostingErr**. Read-write. UART #1 AHB to APB write posting with error.
24	**Uart0RegsH2PPostingErr**. Read-write. UART #0 AHB to APB write posting with error.
23:19	Reserved.
18	**H2X2RegsRspTimedOut**. Read-write. Upstream AHB to AXI #2 write response or read data timed out.
17	**H2X2RegsWriteDataStarved**. Read-write. Upstream AHB to AXI #2 write data starved.
16	**H2X2RegsReadPurged**. Read-write. Upstream AHB to AXI #2 to read prefetch data purged.
15:11	Reserved.
10	**H2X1RegsRspTimedOut**. Read-write. Upstream AHB to AXI #1 write response or read data timed out.
9	**H2X1RegsWriteDataStarved**. Read-write. Upstream AHB to AXI #1 write data starved.
8	**H2X1RegsReadPurged**. Read-write. Upstream AHB to AXI #1 to read prefetch data purged.
7:3	Reserved.
2	**H2X0RegsRspTimedOut**. Read-write. Upstream AHB to AXI #0 write response or read data timed out.
1	**H2X0RegsWriteDataStarved**. Read-write. Upstream AHB to AXI #0 write data starved.
0	**H2X0RegsReadPurged**. Read-write. Upstream AHB to AXI #0 to read prefetch data purged.

AL2AHBx60 AMBA Mask1

Reset: 0000_0000h.

Bits	Description
31:18	Reserved.
17	**X2XRegsNonModErrMask**. Read-write. Mask bit for AL2AHBx50[X2XRegsNonModErr]. 1=Mask.
16	**X2XRegsBufRespErrMask**. Read-write. Mask bit for AL2AHBx50[X2XRegsBufRespErr]. 1=Mask.
15:0	Reserved.

AL2AHBx64 AMBA Mask2

Reset: 0000_0000h.

Bits	Description
31:30	Reserved.
29	**I2C1RegsH2PPostingErrMask**. Read-write. Mask bit for AL2AHBx54[I2C1RegsH2PPostingErr]. 1=Mask.
28	**I2C0RegsH2PPostingErrMask**. Read-write. Mask bit for AL2AHBx54[I2C0RegsH2PPostingErr]. 1=Mask.
27:26	Reserved.
25	**Uart1RegsH2PPostingErrMask**. Read-write. Mask bit for AL2AHBx54[Uart1RegsH2PPostingErr]. 1=Mask.
24	**Uart0RegsH2PPostingErrMask**. Read-write. Mask bit for AL2AHBx54[Uart0RegsH2PPostingErr]. 1=Mask.
23:19	Reserved.
18	**H2X2RegsRspTimedOutMask**. Read-write. Mask bit for AL2AHBx54[H2X2RegsRspTimedOut]. 1=Mask.
17	**H2X2RegsWriteDataStarvedMask**. Read-write. Mask bit for AL2AHBx54[H2X2RegsWriteDataStarved]. 1=Mask.
16	**H2X2RegsReadPurgedMask**. Read-write. Mask bit for AL2AHBx54[H2X2RegsReadPurged]. 1=Mask.
15:11	Reserved.
10	**H2X1RegsRspTimedOutMask: H2X1 Regs rsp timed out mask**. Read-write. Mask bit for AL2AHBx54[H2X1RegsRspTimedOut]. 1=Mask.
9	**H2X1RegsWriteDataStarvedMask**. Read-write. Mask bit for AL2AHBx54[H2X1RegsWriteDataStarved]. 1=Mask.
8	**H2X1RegsReadPurgedMask**. Read-write. Mask bit for AL2AHBx54[H2X1RegsReadPurged]. 1=Mask.
7:3	Reserved.
2	**H2X0RegsRspTimedOutMask**. Read-write. Mask bit for AL2AHBx54[H2X0RegsRspTimedOut]. 1=Mask.
1	**H2X0RegsWriteDataStarvedMask**. Read-write. Mask bit for AL2AHBx54[H2X0RegsWriteDataStarved]. 1=Mask.
0	**H2X0RegsReadPurgedMask**. Read-write. Mask bit for AL2AHBx54[H2X0RegsReadPurged]. 1=Mask.

3.26.19 DMA Registers

There are two DMA controllers in this FCH. The DMA registers are memory-mapped. DMA controller 0 registers range from FEDC_7000h to FEDC_7FFFh. DMA Controller 1 registers range from FEDC_9000h to FEDC_9FFFh.

DMA[1:0]x000 DMA Manager Status Register (DSR)

Bits	Description
31:10	Reserved.
9	**DNS**. Read-only. Reset: 0. Specifies the security status of the DMA manager thread. 0=DMA manager operates in the Secure state. 1=DMA manager operates in the Non-secure state. Note: You must use the boot_manager_ns tie-off signal to set the secure state of the DMA manager thread.
8:4	**WakeupEvent**. Read-only. Reset: 0. When the DMA manager thread executes a DMAWFE instruction, it waits for the following event to occur: Bits — Definition 00000b — Event[0] 00001b — Event[1] 00010b — Event[2] 11110b-00011b — Event[WakeupEvent] 11111b — Event[31]
3:0	**DmaStatus**. Read-only. Reset: 0. Specifies the operating state of the DMA manager: Bits — Definition 0000b — Stopped 0001b — Executing 0010b — Cache miss 0011b — Updating PC 0100b — Waiting for event 1110b-0101b — Reserved 1111b — Fault

DMA[1:0]x004 DMA Program Counter Register(DPC)

Bits	Description
31:0	**PcMgr**. Read-only. Reset: 0. Program counter for the DMA manager thread.

DMA[1:0]x020 Interrupt Enable Register (INTEN)

Bits	Description
31:0	**EventIrqSelect**. Read-write. Reset: 0. Program the appropriate bit to control how the DMA controller responds when it executes DMASEV. 0=If the DMAC executes DMASEV for the event-interrupt resource N then the DMAC signals event N to all of the threads. Set bit[N] to 0 if your system design does not use IRQ[N] to signal an interrupt request. 1=If the DMAC executes DMASEV for the event-interrupt resource N then the DMAC sets IRQ[N] HIGH. Set bit[N] to 1 if your system design requires IRQ[N] to signal an interrupt request.

DMA[1:0]x024 Event-Interrupt Raw Status Register (INT_EVENT_RIS)

Bits	Description
31:0	**DMASEVActive**. Read-only. Reset: 0. Returns the status of the event-interrupt resources: 0=Event N is inactive or IRQ[N] is LOW. 1=Event N is active or IRQ[N] is High. Note: When the DMAC executes a DMASEV N instruction to send event N, the INTEN Register controls whether the DMAC: • Signals an interrupt using the appropriate IRQ. • Sends the event to all of the threads. Note: The DMAC clears bit[N] when either: • The INTEN Register is programmed to process the event and the DMAC executes a DMAWFE instruction for that event. • The INTEN Register is programmed to signal an interrupt and you write the corresponding bit to the INTCLR Register.

DMA[1:0]x028 Interrupt Status Register (INTMIS)

Bits	Description
31:0	**IrqStatus**. Read-only. Reset: 0. Provides the status of the interrupts that are active in the DMAC. 0=Interrupt N is inactive and therefore IRQ[N] is LOW. 1=Interrupt N is active and therefore IRQ is HIGH. Note: The INTCLR Register is used to set Bit[N] to 0. Note: Bit[N] is 0 if the INTEN Register programs DMASEV to signal an event.

DMA[1:0]x02C Interrupt Clear Register (INTCLR)

Bits	Description
31:0	**IrqClr**. Write-only. Reset: 0. Controls the clearing of the IRQ outputs: 0=The status of IRQ[N] does not change. 1=The DMAC sets IRQ[N] LOW if the INTEN Register programs the DMAC to signal an interrupt. Otherwise, the status of IRQ[N] does not change.

DMA[1:0]x030 Fault Status DMA Manager Register (FSRD)

Bits	Description
31:1	Reserved.
0	**FsMgr**. Read-only. Reset: 0. Provides the fault status of the DMA manager Read as: 0=The DMA manager thread is not in the Faulting state. 1=The DMA manager thread is in the Faulting state.

DMA[1:0]x034 Fault Status DMA Channel Register (FSRC)

Bits	Description
31:8	Reserved.
7:0	**FaultStatus**. Read-only. Reset: 0. Each bit provides the fault status of the corresponding channel Read as: 0=No faults is present on DMA channel N. 1=DMA channel N is in the Faulting or Faulting completing state.

DMA[1:0]x038 Fault Type DMA Manager Register (FTRD)

Bits	Description
31	Reserved.
30	**DbgInstr**. Read-only. Reset: 0. If the DMA manager aborts, this bit indicates whether the erroneous instruction was read from the system memory or from the debug interface: 0=Instruction that generated an abort was read from system memory. 1=Instruction that generated an abort was read from the debug interface.
29:17	Reserved.
16	**InstrFetchErr.** Read-only. Reset: 0. Indicates the AXI response that the DMAC receives on the RRESP bus, after the DMA manager performs an instruction fetch: 0=OKAY response. 1=ECOKAY, SLVERR, or DECERR response.
15:6	Reserved.
5	**MgrEventErr**. Read-only. Reset: 0. Indicates whether the DMA manager was attempting to execute DMAWFE or DMASEV with inappropriate security permissions: 0=The DMA manager has appropriate security to execute DMAWFE or DMASEV. 1=A DMA manager thread in the Non-secure state attempted to execute either: • DMAWFE to wait for a secure event. • DMASEV to create a secure event or secure interrupt.
4	**DmagoErr**. Read-only. Reset: 0. Indicates whether the DMA manager was attempting to execute DMAGO with inappropriate security permissions: 0=The DMA manager has appropriate security to execute DMAGO. 1=A DMA manager thread in the Non-secure state attempted to execute DMAGO to create a DMA channel operating in the Secure state.
3:2	Reserved.
1	**OperandInvalid**. Read-only. Reset: 0. Indicates whether the DMA manager was attempting to execute an instruction operand that was not valid for the configuration of the DMAC: 0=Valid operand. 1=Invalid operand.
0	**UndefInstr**. Read-only. Reset: 0. Indicates whether the DMA manager was attempting to execute an undefined instruction: 0=Defined instruction. 1=Undefined instruction.

DMA[1:0]x0[5C:40:Step4] Fault Type Register

Table 284: Register Mapping for DMA[1:0]x0[5C:40:Step4]

Register	Function
DMA[0]x0[40]	DMA 0 Channel [0] (FTR[0][0])
DMA[0]x0[44]	DMA 0 Channel [1] (FTR[0][1])
DMA[0]x0[48]	Reserved
DMA[0]x0[4C]	Reserved
DMA[0]x0[50]	Reserved
DMA[0]x0[54]	Reserved
DMA[0]x0[58]	Reserved
DMA[0]x0[5C]	Reserved
DMA[1]x0[40]	DMA 1 Channel [0] (FTR[1][0])
DMA[1]x0[44]	DMA 1 Channel [1] (FTR[1][1])
DMA[1]x0[48]	Reserved
DMA[1]x0[4C]	Reserved
DMA[1]x0[50]	Reserved
DMA[1]x0[54]	Reserved
DMA[1]x0[58]	Reserved
DMA[1]x0[5C]	Reserved

Bits	Description
31	**LockupErr**. Read-only. Reset: 0. Indicates whether the DMA channel has locked-up because of resource starvation. This fault is an imprecise abort. 0=DMA channel has adequate resources. 1=DMA channel has locked-up because of insufficient resources.
30	**DbgInstr**. Read-only. Reset: 0. If the DMA channel aborts, this bit indicates whether the erroneous instruction was read from the system memory or from the debug interface. This fault is an imprecise abort but the bit is only valid when a precise abort occurs. 0=Instruction that generated an abort was read from system memory. 1=Instruction that generated an abort was read from the debug interface.
29:19	Reserved.
18	**DataReadErr**. Read-only. Reset: 0. Indicates the AXI response that the DMAC receives on the RRESP bus, after the DMA channel thread performs a data read. This fault is an imprecise abort. 0=OKAY response. 1=EXOKAY, SLVERR, or DECERR response.
17	**DataWriteErr**. Read-only. Reset: 0. Indicates the AXI response that the DMAC receives on the BRESP bus, after the DMA channel thread performs a data write. This fault is an imprecise abort. 0=OKAY response. 1=EXOKAY, SLVERR, or DECERR response.

16	**InstrFetchErr**. Read-only. Reset: 0. Indicates the AXI response that the DMAC receives on the RRESP bus, after the DMA channel thread performs an instruction fetch. This fault is a precise abort. 0=OKAY response. 1=EXOKAY, SLVERR, or DECERR response.
15:14	Reserved.
13	**StDataUnavailable**. Read-only. Reset: 0. Indicates whether the MFIFO did not contain the data to enable the DMAC to perform the DMAST. This fault is a precise abort. 0=MFIFO contains all the data to enable the DMAST to complete. 1=Previous DMALDs have not put enough data in the MFIFO to enable the DMAST to complete.
12	**MfifoErr**. Read-only. Reset: 0. Indicates whether the MFIFO prevented the DMA channel thread from executing DMALD or DMAST instruction. This fault is an imprecise abort. 0=If DMALD, MFIFO contains sufficient space; If DMAST, MFIFO contains sufficient data. 1=If DMALD, MFIFO is too small to hold the data; If DMAST, MFIFO is too small to store the data to enable DMAST to complete.
11:8	Reserved.
7	**ChRdwrErr**. Read-only. Reset: 0. Indicates whether a DMA channel thread, in the non-secure state, attempts to program the CCRn Register to perform a secure read or secure write. This fault is a precise abort. 0=A DMA channel thread in the non-secure state is not violating the security permissions. 1=A DMA channel thread in the non-secure state attempted to perform a secure read or secure write.
6	**ChPeriphErr**. Read-only. Reset: 0. Indicates whether a DMA channel thread, in the non-secure state, attempts to execute DMAWFP, DMALDP, DMASTP, DMAFLUSHP with inappropriate security permissions. This fault is a precise abort. 0=A DMA channel thread in the non-secure state is not violating the security permissions. 1=A DMA channel thread in the non-secure state attempted to execute either: • DMAWFP to wait for a secure peripheral. • DMALDP or DMASTP to notify a secure peripheral. • DMAFLUSHP to flush a secure peripheral.
5	**ChEvntErr**. Read-only. Reset: 0. Indicates whether a DMA channel thread attempts to execute DMAWFE or DMASEV with inappropriate security permissions. This fault is a precise abort. 0=A DMA channel thread in the Non-secure state is not violating the security permissions. 1=A DMA channel thread in the Non-secure state attempted to execute either: • DMAWFE to wait for a secure event. • DMASEV to create a secure event or secure interrupt.
4:2	Reserved.
1	**OperandInvalid**. Read-only. Reset: 0. Indicates whether the DMA channel thread was attempting to execute an instruction operand that was not valid for the configuration of the DMAC. This fault is a precise abort. 0=Valid operand. 1=Invalid operand.
0	**UndefInstr.** Read-only. Reset: 0. Indicates whether the DMA channel thread was attempting to execute an undefined instruction. This fault is a precise abort. 0=Defined instruction. 1=Undefined instruction.

DMA[1:0]x1[08:00:Step8] Channel [1:0] Status Register (CSR[1:0])

Bits	Description
31:22	Reserved.
21	**CNS**. Read-only. Reset: 0. The channel non-secure bit provides the security of the DMA channel: 0=DMA channel operates in the Secure state. 1=DMA channel operates in the Non-secure state. Note: The DMAGO instruction determines the security state of a DMA channel.
20:16	Reserved.
15	**DmawfpPeriph**. Read-only. Reset: 0. When the DMA channel thread executes DMAWFP, this bit indicates whether the periph operand was set: 0=DMAWFP executed with the periph operand not set. 1=DMAWFP executed with the periph operand set.
14	**DmawfpBns**. Read-only. Reset: 0. When the DMA channel thread executes DMAWFP, this bit indicates whether the burst or single operand was set: 0=DMAWFP executed with the periph operand not set. 1=DMAWFP executed with the periph operand set.
13:9	Reserved.
8:4	**WakeupNumber**. Read-only. Reset: 0. If the DMA channel is in the Waiting for event state, or the Waiting for peripheral state, then these bits indicate the event or peripheral number that the channel is waiting for: Bits — Definition 00000b — DMA channel is waiting for event, or peripheral 0. 00001b — DMA channel is waiting for event, or peripheral 1. 00010b — DMA channel is waiting for event, or peripheral 2. 11110b-00101b — DMA channel is waiting for event, or peripheral [WakeupNumber]. 11111b — DMA channel is waiting for event, or peripheral 31.
3:0	**ChannelStatus**. Read-only. Reset: 0. Specifies the channel status encoding. Bits — Definition 0000b — Stopped 0001b — Executing 0010b — Cache miss 0011b — Updating PC 0100b — Waiting for event 0101b — At barrier 0110b — Reserved 0111b — Waiting for peripheral 1000b — Killing 1001b — Completing 1101b-1010b — Reserved 1110b — Faulting completing 1111b — Faulting

DMA[1:0]x1[0C:04:Step8] Channel [1:0] Program Counter Register (CPC[1:0])

Bits	Description
31:0	**PcChnl**. Read-only. Reset: 0. Program counter for the DMA channel[n] thread, where n depends on the address of the register.

DMA[1:0]x4[20:00:Step32] Channel [1:0] Source Address Register (SAR[1:0])

Bits	Description
31:0	**SrcAddr**. Read-only. Reset: 0. Address of the source data for DMA channel[n], where n depends on the address of the register.

DMA[1:0]x4[24:04:Step32] Channel [1:0] Destination Address Register (DAR[1:0])

Bits	Description
31:0	**DstAddr**. Read-only. Reset: 0. Address for the destination data for DMA channel[n], where n depends on the address of the register.

DMA[1:0]x4[28:08:Step32] Channel [1:0] Control Register (CCR[1:0])

Bits	Description
31	Reserved.
30:28	**EndianSwapSize**. Read-only. Reset: 0. Endian swap size: Bits — Definition 000b — No swap, 8-bit data 001b — Swap bytes within 16-bit data 010b — Swap bytes within 32-bit data 011b — Swap bytes within 64-bit data 100b — Swap bytes within 128-bit data 111b-101b — Reserved
27:25	**DstCacheCntl**. Read-only. Reset: 0. Specifies the state of AWCACHE[3] and AWCACHE[1] when the DMAC writes the destination data. AWCACHE[2] is tied low by the DMAC. Setting AWCACHE[3]=1 and AWCACHE[1]=0 violates the AXI protocol. Bit — Definition [2] — 0=AWCACHE[3] is LOW. 1=AWCACHE[3] is HIGH. [1] — 0=AWCACHE[1] is LOW. 1=AWCACHE[1] is HIGH. [0] — 0=AWCACHE[0] is LOW. 1=AWCACHE[0] is HIGH.

24:22	**DstProtCntl**. Read-only. Reset: 010b. Specifies the state of AWPROT[2:0] when the DMAC writes the destination data. Bit — Definition [2] — 0=AWPROT[2] is LOW. 1=AWPROT[2] is HIGH. [1] — 0=AWPROT[1] is LOW. 1=AWPROT[1] is HIGH. [0] — 0=AWPROT[0] is LOW. 1=AWPROT[0] is HIGH. Note: Only DMA channels in the secure state can program AWPROT[1] LOW, that is, a secure access. If a DMA channel in the non-secure state attempts to set AWPROT[1] LOW, then the DMA channel aborts.
21:18	**DstBurstLen**. Read-only. Reset: 0. Specifies the number of data transfers that the DMAC performs when it writes the destination data: Bits — Definition 0000b — 1 data transfer 0001b — 2 data transfers 0010b — 3 data transfers 1110b-0011b — [DstBurstLen+1] data transfers 1111b — 16 data transfers The total number of bytes that the DMAC writes out the MFIFO when it executes a DMAST instructions is the product of DstBurstLen and DstBurstSize. Note: These bits control the state of AWLEN[3:0].
17:15	**DstBurstSize**. Read-only. Reset: 0. For each beat within a burst, it programs the number of bytes that the DMAC writes to the destination: Bits — Definition 000b — Write 1 byte per beat 001b — Write 2 bytes per beat 010b — Write 4 bytes per beat 011b — Write 8 bytes per beat 100b — Write 16 bytes per beat 111b-101b — Reserved The total number of bytes that the DMAC writes out of the MFIFO when it executes a DMAST instruction is the product of DstBurstLen and DstBurstSize. Note: These bits control the state of AWSIZE[2:0].
14	**DstInc**. Read-only. Reset: 0. Program the burst type that the DMAC performs when it writes the destination data: 0=Fixed-address burst; The DMAC signals AWBURST[0] LOW. 1=Incrementing-address burst; The DMAC signals AWBURST[0] HIGH.
13:11	**SrcCacheCtrl**. Read-only. Reset: 0. Set the bus to control the state of ARCACHE[2:0] when the DMAC reads the source data. Bit — Definition [2] — 0=ARCACHE[2] is LOW. 1=ARCACHE[2] is HIGH. [1] — 0=ARCACHE[1] is LOW. 1=ARCACHE[1] is HIGH. [0] — 0=ARCACHE[0] is LOW. 1=ARCACHE[0] is HIGH. Note: The DMAC ties ARCACHE[3] low. Setting ARCACHE[2:1]=10b violates the AXI protocol.

10:8	**SrcProtCtrl**. Read-only. Reset: 010b. Programs the state of ARPROT[2:0] when the DMAC reads the source data. Bit — Definition [2] — 0=ARPROT[2] is LOW. 1=ARPROT[2] is HIGH. [1] — 0=ARPROT[1] is LOW. 1=ARPROT[1] is HIGH. [0] — 0=ARPROT[0] is LOW. 1=ARPROT[0] is HIGH. Note: Only DMA channels in the Secure state can program ARPROT[1] LOW, that is, a secure access. If a DMA channel is the Non-secure state attempts to set ARPROT[1] LOW, the DMA channel aborts.
7:4	**SrcBurstLen**. Read-only. Reset: 0. For each burst, these bits program the number of data transfers that the DMAC performs when it reads the source data: Bits — Definition 0000b — 1 data transfer 0001b — 2 data transfers 0010b — 3 data transfers 1110b-0011b — [SrcBusrtLen+1] data transfers 1111b — 16 data transfers The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of SrcBurstLen and SrcBurstSize. Note: These bits control the state of ARLEN[3:0].
3:1	**SrcBurstSize**. Read-only. Reset: 0. For each beat within a burst, it programs the number of bytes that the DMAC reads from the source: Bits — Definition 000b — Read 1 byte per beat 001b — Read 2 bytes per beat 010b — Read 4 bytes per beat 011b — Read 8 bytes per beat 100b — Read 16 bytes per beat 111b-101b — Reserved The total number of bytes that the DMAC reads into the MFIFO when it executes a DMALD instruction is the product of SrcBurstLen and SrcBurstSize. Note: These bits control the state of ARSIZE[2:0].
0	**SrcInc**. Read-only. Reset: 0. Program the burst type that the DMAC performs when it reads the source data. 0=Fixed-address burst; The DMAC signals ARBURST[0] LOW. 1=Incrementing-address burst; The DMAC signals ARBURST[0] HIGH.

DMA[1:0]x4[2C:0C:Step32] Channel [1:0] Loop Counter 0 Register (LC0[1:0])

Bits	Description
31:8	Reserved.
7:0	**LoopCounter0Iterations**. Read-only. Reset: 0. Specifies the number of loop counter iterations.

DMA[1:0]x4[30:10:Step32] Channel [1:0] Loop Counter 1 Register (LC1[1:0])

Bits	Description
31:8	Reserved.
7:0	**LoopCounter1Iterations**. Read-only. Reset: 0. Specifies the number of loop counter iterations.

DMA[1:0]xD00 Debug Status Register (DBGSTATUS)

Bits	Description
31:1	Reserved.
0	**BdgStatus**. Read-only. Reset: 0. Specifies the debug status. 0=Idle. 1=Busy.

DMA[1:0]xD04 Debug Command Register (DBGCMD)

Bits	Description
31:8	Reserved. Write as zero.
7:0	**DbgCmd**. Write-only. Reset: X. The debug encoding is as follow: Bits — Definition 00b — Execute the instruction that the DBGINST[1:0] Registers contain. 11b-01b — Reserved.

DMA[1:0]xD08 Debug Instruction 0 Register (DBGINST0)

Bits	Description
31:24	**InstructionByte1**. Write-only. Reset: X. Instruction byte 1.
23:16	**InstructionByte0**. Write-only. Reset: X. Instruction byte 0.
15:11	Reserved. Write as zero.
10:8	**ChannelNumber**. Write-only. Reset: X. DMA channel number: Bits — Definition 000b — DMA channel 0 001b — DMA channel 1 111b-010b — Reserved
7:1	Reserved. Write as zero.
0	**DebugThread**. Write-only. Reset: X. The debug thread encoding is as follow: 0=DMA manager thread. 1=DMA channel. Note: When set to 1, the ChannelNumber field selects the DMA channel to debug.

DMA[1:0]xD0C Debug Instruction 1 Register (DBGINST1)

Bits	Description
31:24	**InstructionByte5**. Write-only. Reset: X. Instruction byte 5.

23:16	I**nstructionByte4**. Write-only. Reset: X. Instruction byte 4.
15:8	**InstructionByte3**. Write-only. Reset: X. Instruction byte 3.
7:0	I**nstructionByte2**. Write-only. Reset: X. Instruction byte 2.

DMA[1:0]xE00 Configuration Register 0 (CR0)

Bits	Description
31:22	Reserved.
21:17	**NumEvents**. Read-only. Reset: 0001b. Specifies the number of interrupt outputs that the DMAC provides. Bits — Definition 0000b — Reserved 0001b — 2 interrupt outputs, IRQ[1:0] 1111b-0010b — Reserved
16:12	**NumPeriphReq**. Read-only. Reset: 0001b. Specifies the number of peripheral request interfaces that the DMAC provides. Bits — Definition 0000b — Reserved 0001b — 2 peripheral request interfaces 1111b-0010b — Reserved Note: This field is only valid when the PeriphReq bit is set to 1.
11:7	Reserved.
6:4	**NumChnls**. Read-only. Reset: 001b. Specifies the number of DMA channels that the DMAC supports. Bits — Definition 000b — Reserved 001b — 2 DMA channels 111b-010b — Reserved
3	Reserved.
2	**MgrNsAtRst**. Read-only. Reset: 0. Indicates the status of the boot_manager_ns tie-off signal when the DMAC exited from reset. 0=Boot_manager_ns was LOW. 1=Boot_manager_ns was HIGH.
1	**BootEn**. Read-only. Reset: 0. Indicates the status of the boot_from_pc tie-off signal when the DMAC exited from reset. 0=Boot_from_pc was LOW. 1=Boot_from_pc was HIGH.
0	**PeriphReq**. Read-only. Reset: 0. Supports peripheral requests. 0=The DMAC does not provide a peripheral request interface. 1=The DMAC provides the number of peripheral request interfaces that the NumPeriphReq field specifies.

DMA[1:0]xE04 Configuration Register 1 (CR1)

Bits	Description
31:8	Reserved.
7:4	**NumIcacheLines**. Read-only. Reset: 0111b. Specifies the number of I-cache lines. Bits Definition 0110b-0000b Reserved 0111b 8 I-cache lines 1111b-1000b Reserved
3	Reserved.
2:0	**IcacheLen**. Read-only. Reset: 011b. Specifies the length of an I-cache line. Bits Definition 010b-000b Reserved 011b 8 bytes 111b-100b Reserved

DMA[1:0]xE08 Configuration Register 2 (CR2)

Bits	Description
31:0	**BootAddr**. Read-only. Reset: 0. Provides the value of boot_addr[31:0] tie-off signals when the DMAC exited from reset.

DMA[1:0]xE0C Configuration Register 3 (CR3)

Bits	Description
31:0	**Ins**. Read-only. Reset: X. Provides the security state of an event-interrupt resource: 0=Event[N] or IRQ[N] is in the Secure state. 1=Event[N] or IRQ[N] is in the Non-secure state. Note: The boot_irq_ns[3:0] tie-off signals initialize the bits in this register when the DMAC exits from reset.

DMA[1:0]xE10 Configuration Register 4 (CR4)

Bits	Description
31:0	**Pns**. Read-only. Reset: X. Specifies the security state of the peripheral request interfaces: 0=Peripheral request interface N is in the Secure state. 1=Peripheral request interface N is in the Non-secure state. Note: The boot_periph_ns[3:0] tie-off signals initialize the bits in this register when the DMAC exits from reset.

DMA[1:0]xE14 DMA Configuration Register (CRD)

Bits	Description
31:30	Reserved.
29:20	**DataBufferDep**. Read-only. Reset: 00_0111_1111b. The number of lines that the data buffer contains: Bits — Definition 00_0111_1110b-00_0000_0000b — Reserved 00_0111_1111b — 128 lines 11_1111_1111b-00_1000_0000b — Reserved
19:16	**RdQDep**. Read-only. Reset: 0011b. The depth of the read queue: Bits — Definition 0010b-0000b — Reserved 0011b — 4 lines 1111b-0100b — Reserved
15	Reserved.
14:12	**RdCap**. Read-only. Reset: 001b. Read issuing capability that programs the number of outstanding read transactions: Bits — Definition 000b — Reserved 001b — 2 111b-010b — Reserved
11:8	**WrQDep**. Read-only. Reset: 001b. The depth of the write queue: Bits — Definition 0000b — Reserved 0001b — 2 lines 1111b-0010b — Reserved.
7	Reserved.
6:4	**WrCap**. Read-only. Reset: 0. Write issuing capability that programs the number of outstanding write transactions: Bits — Definition 000b — 1 111b-001b — Reserved.
3	Reserved.
2:0	DataWidth. Read-only. Reset: 010b. The data bus width of the AXI master interface: Bits — Definition 001b-000b — Reserved 010b — 32-bit 111b-011b — Reserved.

DMA[1:0]xE80 Watchdog Register (WD)

Bits	Description
31:1	Reserved.
0	**WdIrqOnly**. Read-write. Reset: X. Control how the DMAC responds when it detects a lock-up condition: 0=The DMAC aborts all the contributing DMA channels and sets irq_abort HIGH. 1=The DMAC sets irq_abort HIGH.

DMA[1:0]xFE0 Peripheral Identification Register 0

Bits	Description
31:8	Reserved.
7:0	**PartNumber0.** Read-only. Reset: X. Specifies the peripheral part number 0.

DMA[1:0]xFE4 Peripheral Identification Register 1

Bits	Description
31:8	Reserved.
7:4	**Designer0**. Read-only. Reset: X. Specified the designer 0.
3:0	**PartNumber1.** Read-only. Reset: 3h. Specifies the peripheral part number 1.

DMA[1:0]xFE8 Peripheral Identification Register 2

Bits	Description
31:8	Reserved.
7:4	**Revision**. Read-only. Reset: X. Identifies the revision.
3:0	**Designer1**. Read-only. Reset: X. Specifies the designer 1.

DMA[1:0]xFEC Peripheral Identification Register 3

Bits	Description
31:1	Reserved.
0	**IntegrationCfg**. Read-only. Reset: 0. 0=The DMA does not contain integration test logic.

DMA[1:0]xFF0 Component Identification Register 0

Bits	Description
31:8	Reserved.
7:0	**PcellId0**. Read-only. Reset: X.

DMA[1:0]xFF4 Component Identification Register 1

Bits	Description
31:8	Reserved.
7:0	**PcellId1**. Read-only. Reset: X.

DMA[1:0]xFF8 Component Identification Register 2

Bits	Description
31:8	Reserved.
7:0	**PcellId2**. Read-only. Reset: X.

DMA[1:0]xFFC Component Identification Register 3

Bits	Description
31:8	Reserved.
7:0	**PcellId3**. Read-only. Reset: X.

3.26.20 I²C Configuration Registers

There are four I²C controller in this FCH. The I²C registers are memory-mapped. The register address map is listed in Table 285 [Address Space for I2C controller].

Table 285: Address Space for I²C controller

Function	Address
I²C Controller 0	FEDC_2XXXh
I²C Controller 1	FEDC_3XXXh
I²C Controller 2	FEDC_4XXXh
I²C Controller 3	FEDC_5XXXh

I2C[3:0]x00 I²C Control Register (IC_CON)

Table 286: States for I2C[3:0]x00[IcSlaveDisable] and I2C[3:0]x00[MasterMode]

I2C[3:0]x00[IcSlaveDisable]	I2C[3:0]x00[MasterMode]	State
0	0	Slave Device
0	1	Config Error
1	0	Config Error
1	1	Master Device

Bits	Description
15:7	Reserved.
6	**IcSlaveDisable**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. Controls whether I²C has its slave disabled. 0=Slave is enabled. 1=Slave is disabled; the controller only functions as a master and does not perform any action that requires a slave. The controller should only be used either as a master or as a slave. See Table 286 [States for I2C[3:0]x00[IcSlaveDisable] and I2C[3:0]x00[MasterMode]] for valid configuration.
5	**IcRestartEn**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. Determines whether RESTART conditions can be sent when acting as a master. 1=Enable. 0=Disable. When the RESTART is disabled, the following operations cannot be performed: • Sending a START BYTE. • Performing any high-speed mode operation. • Performing direction changes in combined format mode. • Performing a read operation with a 10-bit address. Performing the above operations results in setting bit I2C[3:0]x34[TxAbrt].
4	**Ic10BitAddrMaster**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. Controls whether the I²C controller starts its transfers in 7-bit or 10-bit addressing mode when acting as a master. 1=10-bit addressing. 0=7-bit addressing.

3	**Ic10BitAddrSlave**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. Controls whether the I²C controller responds to 7-bit or 10-bit addresses when acting as a slave. 0=7-bit addressing; the controller ignores transactions that involve 10-bit addressing; for 7-bit addressing, only the lower 7 bits of I2C[3:0]x04[IcTar] are compared. 1=10-bit addressing; the controller responds to only 10-bit addressing transfers that match the full 10 bits of I2C[3:0]x04[IcTar].
2:1	**Speed**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 3. Controls the speed that the I²C controller operates at in master mode. Only values in the range of 1 to I2C[3:0]xF4[MaxSpeedMode] are valid. Programming this field with invalid speed results in the value of I2C[3:0]xF4[MaxSpeedMode]. Bits Definition 00b Reserved. 01b Standard speed mode (0 to 100 kbit/s). 10b Fast speed mode (<= 400 kbit/s) or fast mode plus (≤ 1000 kbit/s). 11b Reserved.
0	**MasterMode**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. Controls whether the I²C master is enabled. 1=Master is Enabled. The controller should only be used either as a master or as a slave. See Table 286 [States for I2C[3:0]x00[IcSlaveDisable] and I2C[3:0]x00[MasterMode]] for valid configuration.

I2C[3:0]x04 I²C Target Address Register (IC_TAR)

It is not necessary to perform any write to this register if the I²C controller is enabled as an I²C slave only.

Bits	Description
15:12	Reserved.
11	**Special**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. Indicates whether the software performs a General Call or START BYTE command. 0=Ignore the GcOrStart bit and use the I2C[3:0]x04[IcTar] register normally. 1=Perform the special I²C command as specified in the GcOrStart bit.
10	**GcOrStart: Genaral Call or Start**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0. IF (I2C[3:0]x04[Special] == 0) THEN Ignore this bit. ELSE Indicates whether the I²C controller is to perform a General Call or START byte command. 1=START BYTE. 0=General Call; After issuing a General Call command, only writes may be performed; Attempting to issue a read command results in setting I2C[3:0]x34[TxAbrt]; The controller remains in General Call mode until I2C[3:0]x04[Special] == 0. ENDIF.
9:0	**IcTar: I²C Target Address Register**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 55h. Specifies the target address for any master transactions. When the controller transmits a General Call, these bits are ignored. To generate a START BYTE, the CPU needs to write only once into these bits. Address locations at 07h-00h or 7Fh-78h are reserved. Do not program I2C[3:0]x08[IcSar] or I2C[3:0]x04[IcTar] to a reserved value. If (I2C[3:0]x04[IcTar] == I2C[3:0]x08[IcSar]), loopback exists. Since the FIFOs are shared between master and slave. full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself; it can transmit to only a slave.

I2C[3:0]x08 I²C Slave Address Register (IC_SAR)

It is not necessary to perform any write to this register if the I²C controller is enabled as an I²C master only.

Bits	Description
15:10	Reserved.
9:0	**IcSar: I²C Slave Address Register**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 55h. Specifies the slave address when the I²C operates as a slave. For 7-bit addressing, only IcSar[6:0] is used. Address locations at 07h-00h or 7Fh-78h are reserved. Do not program I2C[3:0]x08[IcSar] or I2C[3:0]x04[IcTar] to a reserved value.

I2C[3:0]x10 I²C RX/TX Data Buffer and Command Register (IC_DATA_CMD)

The CPU writes to this register when filling TX FIFO. The CPU reads from this register when retrieving bytes from RX FIFO. In order for the I²C controller to continue acknowledging reads, a read command is to be written for every byte that is to be received; otherwise, the controller stops acknowledging.

Bits	Description
15:11	Reserved.
10	**Restart**. Write-only. Reset: X. Controls whether a RESTART is issued before the byte is sent or received. Bits — Definition 1b — If (I2C[3:0]x00[IcRestartEn] == 1), a RESTART is issued before the data is sent or received, regardless of whether or not the transfer direction changes from the previous command; If (I2C[3:0]x00[IcRestartEn] == 0), a STOP, followed by a START is issued instead. 0b — If (I2C[3:0]x00[IcRestartEn] == 1), a RESTART is issued only if the transfer direction changes from the previous command; If (I2C[3:0]x00[IcRestartEn] == 0), a STOP, followed by a START is issued instead.
9	**Stop**. Write-only. Reset: X. Controls whether a STOP is issued after the byte is sent or received. Bits — Definition 1b — STOP is issued after this byte, regardless of whether or not the TX FIFO is empty. If the TX FIFO is not empty, the master immediately tries to start a new transfer by issuing a START and arbitrating for the bus. 0b — STOP is not issued after this byte, regardless of whether or not the TX FIFO is empty. If the TX FIFO is not empty, the master continues the current transfer by sending and receiving data bytes according to the value of I2C[3:0]x10[Cmd]. If the TX FIFO is empty, the master holds the SCL line low and stalls the bus until a new command is available in the TX FIFO.
8	**Cmd: Command**. Write-only. Reset: 0. Controls whether a read or a write is performed. It only controls the direction when the controller acts as a master. 1=Read. 0=Write. In slave-receiver mode, this bit is ignored because writes to this register are not required. In slave-transmitter mode, a “0” indicates that the data in I2C[3:0]x10[Dat] is to be transmitted.
7:0	**Dat: Data**. Read-write. Reset: 0. Contains the data to be transmitted or received on the I²C bus.

I2C[3:0]x14 Standard Speed I²C Clock SCL High Count Register (IC_SS_SCL_HCNT)

Bits	Description
15:0	**IcSsSclHcnt: I²C Standard Speed SCL High Count**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0190h. Sets the SCL clock high-period count for standard speed mode. This register must be programmed before any I²C bus transaction can take place to ensure proper IO timing. The minimum valid value is 6; the hardware prevents values less than this being written, and if attempted, results in 6 being set. This register must not be programmed to a value higher than 65525.

I2C[3:0]x18 Standard Speed I²C Clock SCL Low Count Register (IC_SS_SCL_LCNT)

Bits	Description
15:0	**IcSsSclLcnt: I²C Standard Speed SCL Low Count**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 01D6h. Sets the SCL clock low-period count for standard speed mode. This register must be programmed before any I²C bus transaction can take place to ensure proper IO timing. The minimum valid value is 6; the hardware prevents values less than this being written, and if attempted, results in 6 being set.

I2C[3:0]x1C Fast Speed I²C Clock SCL High Count Register (IC_FS_SCL_HCNT)

Bits	Description
15:0	**IcFsSclHcnt: I²C Fast Speed SCL High Count**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 003Ch. Sets the SCL clock high-period count for fast speed mode. This register must be programmed before any I²C bus transaction can take place to ensure proper IO timing. The minimum valid value is 6; the hardware prevents values less than this being written, and if attempted, results in 6 being set.

I2C[3:0]x20 Fast Speed I²C Clock SCL Low Count Register (IC_FS_SCL_LCNT)

Bits	Description
15:0	**IcFsSclLcnt: I²C Fast Speed SCL Low Count**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0082h. Sets the SCL clock low-period count for fast speed mode. This register must be programmed before any I²C bus transaction can take place to ensure proper IO timing. The minimum valid value is 8; the hardware prevents values less than this being written, and if attempted, results in 8 being set.

I2C[3:0]x24 High Speed I²C Clock SCL High Count Register (IC_HS_SCL_HCNT)

Bits	Description
15:0	**IcHsSclHCnt: I²C[3:0] High Speed SCL High Count**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0006h. Sets the SCL clock high-period count for high speed mode. This register must be programmed before any I²C bus transaction can take place to ensure proper IO timing. The SCL High time depends on the loading of the bus. For 100 pF loading, the SCL High time is 60 ns; for 400 pF loading, the SCL High time is 120 ns. The minimum valid value is 6; the hardware prevents values less than this being written, and if attempted, results in 6 being set.

I2C[3:0]x28 High Speed I²C Clock SCL Low Count Register (IC_HS_SCL_LCNT)

Bits	Description
15:0	**IcHsSclLcnt: I²C High Speed SCL Low Count**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 0010h. Sets the SCL clock low-period count for high speed mode. This register must be set before any I²C bus transaction can take place to ensure proper IO timing. The SCL low time depends on the loading of the bus. For 100 pF loading, the SCL low time is 160 ns; for 400 pF loading, the SCL low time is 320 ns. The minimum valid value is 8; the hardware prevents values less than this being written, and if attempted, results in 8 being set.

I2C[3:0]x2C I²C Interrupt Status Register (IC_INTR_STAT)

Reset: 0000h. This register is a masked version of the raw status defined in I2C[3:0]x34. Each bit in this register has a corresponding mask bit in I2C[3:0]x30. These bits are cleared by reading the matching interrupt clear register.

Bits	Description
15:12	Reserved.
11	**RGenCall**. Read-only. Masked version of I2C[3:0]x34[GenCall].
10	**RStartDet**. Read-only. Masked version of I2C[3:0]x34[StartDet].
9	**RStopDet**. Read-only. Masked version of I2C[3:0]x34[StopDet].
8	**RActivity**. Read-only. Masked version of I2C[3:0]x34[Activity].
7	**RRxDone**. Read-only. Masked version of I2C[3:0]x34[RxDone].
6	**RTxAbrt**. Read-only. Masked version of I2C[3:0]x34[TxAbrt].
5	**RRdReq**. Read-only. Masked version of I2C[3:0]x34[RdReq].
4	**RTxEmpty**. Read-only. Masked version of I2C[3:0]x34[TxEmpty].
3	**RTxOver**. Read-only. Masked version of I2C[3:0]x34[TxOver].
2	**RRxFull**. Read-only. Masked version of I2C[3:0]x34[RxFull].
1	**RRxOver**. Read-only. Masked version of I2C[3:0]x34[RxOver].
0	**RRxUnder**. Read-only. Masked version of I2C[3:0]x34[RxUnder].

I2C[3:0]x30 I²C Interrupt Mask Register (IC_INTR_MASK)

Reset: 8FFh. These bits mask their corresponding interrupt status bits. The unmasked raw versions and detailed descriptions of these bits are available in I2C[3:0]x34. The masked status bits are in I2C[3:0]x2C.

Bits	Description
15:12	Reserved.
11	**MGenCall**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
10	**MStartDet**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
9	**MStopDet**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
8	**MActivity**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
7	**MRxDone**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
6	**MTxAbrt**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
5	**MRdReq**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
4	**MTxEmpty**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
3	**MTxOver**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
2	**MRxFull**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
1	**MRxOver**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.
0	**MRxUnder**. Read-write. 0=Mask the interrupt. 1=Unmask the interrupt.

I2C[3:0]x34 I²C Raw Interrupt Status Register (IC_RAW_INTR_STAT)

Reset: 0000h. Unlike the I2C[3:0]x2C register, these bits are not masked so they always show the true status of the I²C controller. These bits are cleared by reading the matching interrupt clear registers located at I2C[3:0]x[68:40].

Bits	Description
15:12	Reserved.
11	**GenCall**. Read-only. 1=A General Call address is received and acknowledged.
10	**StartDet**. Read-only. 1=Either a START or RESTART condition occurred on the I²C interface, regardless of whether the I²C controller is operating in slave or master mode.
9	**StopDet**. Read-only. 1=A STOP condition occurred on the I²C interface, regardless of whether the I²C controller is operating in slave or master mode.
8	**Activity**. Read-only. 1=There was I²C activity. This bit stays set until it is cleared by one of the following: • Disabling the I²C controller. • Reading the I2C[3:0]x5C register. • Reading the I2C[3:0]x40 register. • System reset.
7	**RxDone**. Read-only. 1=The master does not acknowledge a transmitted byte, when the I²C controller acts as a slave-transmitter. This occurs on the last byte of the transmission, indicating that the transmission is done.

6	**TxAbrt**. Read-only. 1=The I²C controller is unable to complete the intended actions on the contents of the transmit FIFO, when acting either as a master transmitter or as a slave transmitter. I2C[3:0]x80 indicates the reason why the transmit abort occurs. The controller flushes, resets, or empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state and won't be ready to accept more data until I2C[3:0]x54 is read.
5	**RdReq**. Read-only. 1=The I²C controller is acting as a slave and another I²C master is attempting to read data from the controller. In this case, the controller holds the I²C bus in a wait state (SCL == 0) until this interrupt is serviced. The processor must respond to this interrupt and write the requested data to I2C[3:0]x10.
4	**TxEmpty**. Read-only; Updated-by-hardware. 1=The transmit buffer is at or below the threshold value set in I2C[3:0]x3C. This bit is automatically cleared by hardware when the buffer level goes above the threshold. When I2C[3:0]x6C[Enable] == 0, the TX FIFO is flushed and held in reset; this bit is 1 when there is activity; this bit is 0 when there is no longer activity (I2C[3:0]x9C[IcEn] == 0).
3	**TxOver**. Read-only. 1=During transmit the transmit buffer is filled to I2C[3:0]xF4[TxBufferDepth] and the software attempts to issue another I²C command by programming I2C[3:0]x10. When I2C[3:0]x6C[Enable] == 0, this bit keeps its level until the master or slave state machines go into idle; this interrupt is cleared when I2C[3:0]x9C[IcEn] == 0.
2	**RxFull**. Read-only; Updated-by-hardware. 1=The receive buffer reaches or goes above the receive threshold specified by I2C[3:0]x38. This bit is automatically cleared by hardware when buffer level goes below the threshold. When I2C[3:0]x6C[Enable] == 0, this bit is cleared regardless of the activity that continues.
1	**RxOver**. Read-only. 1=The receive buffer is completely filled to I2C[3:0]xF4[RxBufferDepth] and an additional byte is received from an external I²C device. The I²C controller acknowledges this, but data bytes that arrive after the FIFO is full are lost. When I2C[3:0]x6C[Enable] == 0, this bit keeps its level until the master or slave state machines go into idle; this interrupt is cleared when I2C[3:0]x9C[IcEn] == 0.
0	**RxUnder**. Read-only. 1=The processor attempts to read the receive buffer when it is empty by reading from I2C[3:0]x10. When I2C[3:0]x6C[Enable] == 0, this bit keeps its level until the master or slave state machines go into idle; this interrupt is cleared when I2C[3:0]x9C[IcEn] == 0.

I2C[3:0]x38 I²C Receive FIFO Threshold Register (IC_RX_TL)

Bits	Description
15:8	Reserved.
7:0	**RxTL: Receive FIFO Threshold Level**. Read-write. Reset: 0. Controls the level of entries (or above) that triggers the I2C[3:0]x34[RxFull] interrupt. The valid range is from 0 to I2C[3:0]xF4[RxBufferDepth]. Attempting to program a value larger than depth of the buffer results in the actual value being the maximum depth of the buffer. Bits — Definition 00h — 1 entry threshold FEh-01h — <RxTL+1> entries threshold FFh — 256 entries threshold

I2C[3:0]x3C I²C Transmit FIFO Threshold Register (IC_TX_TL)

Bits	Description
15:8	Reserved.
7:0	**TxTL: Transmit FIFO Threshold Level**. Read-write. Reset: 0. Controls the level of entries (or below) that triggers the I2C[3:0]x34[TxEmpty] interrupt. The valid range is from 0 to I2C[3:0]xF4[TxBufferDepth]. Attempting to program a value larger than depth of the buffer results in the actual value being the maximum depth of the buffer. Bits — Definition 00h — 0 entry threshold FEh-01h — <TxTL> entries threshold FFh — 255 entries threshold

I2C[3:0]x40 Clear Combined and Individual Interrupt Register (IC_CLR_INTR)

Bits	Description
15:1	Reserved.
0	**ClrIntr**. Read-only. Reset: 0. Read this register to clear the combined interrupt, all individual interrupts, and the I2C[3:0]x80 register. This bit does not clear hardware clearable interrupts, only software clearable interrupts. Refer to I2C[3:0]x80[AbrtSByteNoRstrt] for an exception to clearing I2C[3:0]x80.

I2C[3:0]x44 Clear RxUnder Interrupt Register (IC_CLR_RX_UNDER)

Bits	Description
15:1	Reserved.
0	**ClrRxUnder**. Read-only. Reset: 0. Reading this register clears the I2C[3:0]x34[RxUnder] interrupt.

I2C[3:0]x48 Clear RxOver Interrupt Register (IC_CLR_RX_OVER)

Bits	Description
15:1	Reserved.
0	**ClrRxOver**. Read-only. Reset: 0. Reading this register clears the I2C[3:0]x34[RxOver] interrupt.

I2C[3:0]x4C Clear TxOver Interrupt Register (IC_CLR_TX_OVER)

Bits	Description
15:1	Reserved.
0	**ClrTxOver**. Read-only. Reset: 0. Reading this register clears the I2C[3:0]x34[TxOver] interrupt.

I2C[3:0]x50 Clear RdReq Interrupt Register (IC_CLR_RD_REQ)

Bits	Description
15:1	Reserved.
0	**ClrRdReq**. Read-only. Reset: 0. Reading this register clears the I2C[3:0]x34[RdReq] interrupt.

I2C[3:0]x54 Clear TxAbrt Interrupt Register (IC_CLR_TX_ABRT)

Bits	Description
15:1	Reserved.
0	**ClrTxAbrt**. Read-only. Reset: 0. Reading this register clears I2C[3:0]x34[TxAbrt] and the I2C[3:0]x80 register. This action also releases the TX FIFO from the flushed or reset state, allowing more writes to the TX FIFO. Refer to I2C[3:0]x80[AbrtSByteNoRstrt] for an exception to clearing I2C[3:0]x80.

I2C[3:0]x58 Clear RxDone Interrupt Register (IC_CLR_RX_DONE)

Bits	Description
15:1	Reserved.
0	**ClrRxDone**. Read-only. Reset: 0. Reading this register clears the I2C[3:0]x34[RxDone] interrupt.

I2C[3:0]x5C Clear Activity Interrupt Register (IC_CLR_ACTIVITY)

Reset: 0000h.

Bits	Description
15:1	Reserved.
0	**ClrActivity**. Read-only. Reading this register clears the I2C[3:0]x34[Activity] interrupt if the I^2C is not active anymore. If the I^2C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register gives the status of the I2C[3:0]x34[Activity] interrupt.

I2C[3:0]x60 Clear StopDet Interrupt Register (IC_CLR_STOP_DET)

Reset: 0000h.

Bits	Description
15:1	Reserved.
0	**ClrStopDet: Clr Stop Det**. Read-only. Reading this register clears the I2C[3:0]x34[StopDet] interrupt.

I2C[3:0]x64 Clear StartDet Interrupt Register (IC_CLR_START_DET)

Reset: 0000h.

Bits	Description
15:1	Reserved.
0	**ClrStartDet**. Read-only. Reading this register clears the I2C[3:0]x34[StartDet] interrupt.

I2C[3:0]x68 Clear GenCall Interrupt Register (IC_CLR_GEN_CALL)

Reset: 0000h.

Bits	Description
15:1	Reserved.
0	**ClrGenCall**. Read-only. Reading this register clears the I2C[3:0]x34[GenCall] interrupt.

I2C[3:0]x6C I^2C Enable Register (IC_ENABLE)

Bits	Description
15:2	Reserved.
1	**ABORT**. Read-write. Reset: 0. When set, the controller initiates the transfer abort. 0=ABORT not initiated or ABORT done. 1=ABORT operation in progress. The software can abort the I^2C transfer in master mode by setting this bit. The software can set this bit only when ENABLE is already set; otherwise, the controller ignores any write to ABORT bit. The software cannot clear the ABORT bit once set. In response to an ABORT, the controller issues a STOP and flushes the TX FIFO after completing the current transfer, then sets the TX_ABORT interrupt after the abort operation. The ABORT bit is cleared automatically after the abort operation.
0	**Enable**. Read-write. Reset: 0. Controls whether the I^2C controller is enabled. 1=Enable. 0=Disable. When the controller is disabled, the following actions occur: • The TX FIFO and RX FIFO get flushed. • Status bits in I2C[3:0]x2C and I2C[3:0]x34 are still active until the controller goes into IDLE state (I2C[3:0]x9C[IcEn] == 0). If the module is transmiting, it stops and deletes the contents of the transmit buffer after the current transfer completes. If the module is receiving, the controller stops the current transfer at the end of the current byte and does not acknowledge the transfer. In systems with asynchronous pclk and ic_clk, there is a two ic_clk delay when enabling or disabling the I^2C controller.

I2C[3:0]x70 I^2C Status Register (IC_STATUS)

This register indicates the current transfer status and FIFO status. The status register is readable at any time. None of these bits request an interrupt.

Bits	Description
31:7	Reserved.

6	**SlvActivity: Slave FSM Activity**. Read-only. Reset: 0. Indicates the slave Finite State Machine (FSM) activity status. 0=Slave FSM is in IDLE state; the slave part of the I²C controller is not active. 1=Slave FSM is not in IDLE state; the slave part of the I²C[3:0] controller is active. When I2C[3:0]x9C[IcEn] == 0, this bit is set to 0.
5	**MstActivity: Master FSM Activity**. Read-only. Reset: 0. Indicates master Finite State Machine (FSM) activity status. 0=Master FSM is in IDLE state; the master part of the I²C controller is not active. 1=Master FSM is not in IDLE state; the master part of the I²C[3:0] controller is active. When I2C[3:0]x9C[IcEn] == 0, this bit is set to 0.
4	**RFF: Receive FIFO Completely Full**. Read-only. Reset: 0. 0=Receive FIFO is not full. 1=Recieve FIFO is full. When the I²C is disabled by writing 0 to I2C[3:0]x6C[Enable], this bit is set to 0.
3	**RFNE: Receive FIFO Not Empty**. Read-only. Reset: 0. 0=Recieve FIFO is empty. 1=Recieve FIFO is not empty. When the I²C is disabled by writing 0 to I2C[3:0]x6C[Enable], this bit is set to 0.
2	**TFE: Transmit FIFO Completely Empty**. Read-only. Reset: 1. 0=Transmit FIFO is not empty. 1=Transmit FIFO is empty. When the I²C is disabled by writing 0 to I2C[3:0]x6C[Enable], this bit is set to 1.
1	**TFNF: Transmit FIFO Not Full**. Read-only. Reset:1. 0=Transmit FIFO is full. 1=Transmit FIFO is not full. When the I²C is disabled by writing 0 to I2C[3:0]x6C[Enable], this bit is set to 1.
0	**Activity: I²C Activity Status**. Read-only. Reset: 0. This bit shows the activity status, which is the OR of SlvActivity and MstActivity bits.

I2C[3:0]x74 I²C Transmit FIFO Level Register (IC_TXFLR)

Bits	Description
31:9	Reserved.
8:0	**TxFLR: Transmit FIFO Level**. Read-only; Updated-by-hardware. Reset: 0. Contains the number of valid data entries in the transmit FIFO. The register increments whenever data is placed into the transmit FIFO and decrements when data is retrieved from the transmit FIFO. It is cleared whenever: • The I²C is disabled. • There s a transmit abort (I2C[3:0]x34[TxAbrt] == 1). • A slave bulk transmit mode is aborted.

I2C[3:0]x78 I²C Receive FIFO Level Register (IC_RXFLR)

Bits	Description
31:9	Reserved.
8:0	**RxFLR: Receive FIFO Level**. Read-only; Updated-by-hardware. Reset: 0. Contains the number of valid data entries in the receive FIFO. The register increments whenever data is placed into the receive FIFO and decrements when data is retrieved from the receive FIFO. It is cleared whenever: • The I²C is disabled. • There is a transmit abort caused by any of the events tracked in I2C[3:0]x80.

I2C[3:0]x7C I²C SDA Hold Time Length Register (IC_SDA_HOLD)

Bits	Description
31:16	Reserved.
15:0	**IcSdaHold**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. Controls the amount of hold time on the SDA (Serial Data Line) signal after a negative edge of SCL (Serial Clock) in both master and slave mode, in units of ic_clk period. The value programmed must be greater than the minimum hold time in each mode -- one cycle in master mode, seven cycles in slave mode. The programmed value cannot be larger than N_SCL_LOW-2, where N_SCL_LOW is the duration of the low part of the scl period measured in ic_clk cycles.

I2C[3:0]x80 I²C Transmit Abort Source Register (IC_TX_ABRT_SOURCE)

This register indicates the source of the I2C[3:0]x34[TxAbrt] bit. Except for bit[9], this register is cleared whenever I2C[3:0]x54 or I2C[3:0]x40 is read.

Bits	Description
31:24	**TX_FLUSH_CNT**. Read-only. Reset: 0. This field preserves the TXFLR value prior to the last TX_ABRT event. It is cleared whenever I²C is disabled.
23:17	Reserved.
16	**ABRT_USER_ABRT**. Read-only. Reset: 0. This is a master-mode-only bit. 1=Master has detected the transfer abort (IC_ENABLE[1]).
15	**AbrtSlvRdInTx**. Read-only. Reset: 0. 1=The processor side responds to a slave mode request for data to be transmitted to a remote master and I2C[3:0]x10[Cmd] is written with 1. The I²C controller acts as a slave-transmitter.
14	**AbrtSlvArbLost**. Read-only. Reset: 0. 1=The slave lost the bus while transmitting data to a remote master when the I²C controller acts as a slave-transmitter. Bit[ArbLost] is set at the same time.
13	**AbrtSlvFlushTxFifo**. Read-only. Reset: 0. 1=The slave has received a read command and some data exists in the TX FIFO, so the slave issues a TxAbrt interrupt to flush the old data in TX FIFO. The I²C controller acts as a slave-transmitter.
12	**ArbLost: Arb Lost**. Read-only. Reset: 0. 1=The master has lost arbitration, or the slave transmitter lost arbitration with AbrtSlvArbLost being set. The controller acts as either a slave-transmitter or master-transmitter.
11	**AbrtMasterDis**. Read-only. Reset: 0. 1=The user tries to initiate a master operation with the master mode disabled. The controller acts as either a master-transmitter or master-receiver.
10	**Abrt10BRdNoRstrt**. Read-only. Reset: 0. 1=The master sends a read command in 10-bit addressing mode when I2C[3:0]x00[IcRestartEn] == 0. The controller acts as a master-receiver.

9	**AbrtSByteNoRstrt**. Read-only. Reset: 0. 1=The user tries to send a START byte when I2C[3:0]x00[IcRestartEn] == 0. The controller acts as a master. This bit can be cleared in the same manner as other bits in this registers only after the source of AbrtSByteNoRstrt is fixed as follows: • Restart must be enabled by programming I2C[3:0]x00[IcRestartEn] = 1. • Either clear I2C[3:0]x04[Special] to 0 or clear I2C[3:0]x04[GcOrStart] to 0. Attempting to clear this bit before fixing the source as above will result in this bit being cleared for one cycle and getting re-asserted.
8	**AbrtHsNoRstrt**. Read-only. Reset: 0. 1=The user tries to use the master to transfer data in High Speed mode when I2C[3:0]x00[IcRestartEn] == 0. The controller acts as either a master-transmitter or master-receiver.
7	**AbrtSbyteAckDet**. Read-only. Reset: 0. 1=The master has sent a START byte, and the START byte was acknowledged (wrong behavior). The controller acts as a master.
6	**AbrtHsAckDet**. Read-only. Reset: 0. 1=The master is in High Speed mode and the High Speed Master code was acknowledged (wrong behavior). The controller acts as a master.
5	**AbrtGCallRead**. Read-only. Reset: 0. 1=The controller in master mode sent a General Call, but the user programmed the byte following the General Call to be a read from the bus (I2C[3:0]x10[Cmd] == 1). The controller acts as a master-transmitter.
4	**AbrtGCallNoAck**. Read-only. Reset: 0. 1=The controller in master mode sent a General Call, and no slave on the bus acknowledged the General Call. The controller acts as a master-transmitter.
3	**AbrtTxDataNoAck**. Read-only. Reset: 0. This bit is a master-mode only bit. 1=The master received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledgment from the remote slave(s). The controller acts as a master-transmitter.
2	**Abrt10Addr2NoAck**. Read-only. Reset: 0. 1=The master is in 10-bit address mode, and the second address byte of the 10-bit address was not acknowledged by any slave. The controller acts as either a master-transmitter or master-receiver.
1	**Abrt10Addr1NoAck**. Read-only. Reset: 0. 1=The master is in 10-bit address mode, and the first 10-bit address byte was not acknowledged by any slave. The controller acts as either a master-transmitter or master-receiver.
0	**Abrt7BAddrNoAck**. Read-only. Reset: 0. 1=The master is in 7-bit addressing mode, and the address sent was not acknowledged by any slave. The controller acts as either a master-transmitter or master-receiver.

I2C[3:0]x94 I²C SDA Setup Register (IC_SDA_SETUP)

Bits	Description
31:8	Reserved.
7:0	**SdaSetup**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 64h. Controls the delay measured in ic_clk cycles, that are introduced in the rising edge of SCL (relative to SDA (Serial Data Line) changing) by holding SCL (Serial Clock) low when the controller services a read request while operating as a slave-transmitter. This field must be programmed with a minimum value of 2. The length of setup time = (SdaSetup - 1) * ic_clk_period. For example, if the required delay is 1000 ns, with an ic_clk frequency of 10 MHz, This field should be programmed with a value of 11.

I2C[3:0]x98 I²C ACK General Call Register (IC_ACK_GENERAL_CALL)

Bits	Description
31:1	Reserved.
0	**AckGeneralCall: ACK General Call**. Read-write. Reset: 1. Controls whether the I²C controller responds with a ACK or NACK when it receives an I²C General Call address. 1=The controller responds with a ACK when it receives a General Call. 0=The controller does not generate General Call interrupts.

I2C[3:0]x9C I²C Enable Status Register (IC_ENABLE_STATUS)

The register repots the hardware status when I2C[3:0]x6C[Enable] is programmed from 1 to 0; that is, when the controller is being disabled.

Bits	Description
31:3	Reserved.
2	**SlvRxDataLost: Slave Received Data Lost**. Read-only. Reset: 0. When I2C[3:0]x6C[Enable] == 1, this bit is forced to 0. When I2C[3:0]x6C[Enable] is programmed from 1 to 0, this bit is valid when IcEn == 0; It indicates if a slave-receiver operation has been aborted with at least one data byte received from an I²C transfer due to disabling the I²C controller. 1=The I²C controller is engaged in an aborted I²C transfer (with matching address) and the data phase of the I²C transfer was entered, even though a data byte was responded with a NACK; Or The remote I²C master terminates the transfer with a STOP condition before the I²C controller can NACK a transfer when I2C[3:0]x6C[Enable] is programmed from 1 to 0. 0=The I²C controller is disabled without being actively involved in the data phase of a slave-receiver transfer when I2C[3:0]x6C[Enable] is programmed from 1 to 0.
1	**SlvDisabledWhileBusy**. Read-only. Reset: 0. When I2C[3:0]x6C[Enable] == 1, this bit is forced to 0. When I2C[3:0]x6C[Enable] is programmed from 1 to 0, this bit is valid when IcEn == 0; It indicates if a potential or active slave operation has been aborted due to disabling the I²C controller. This bit is set to 1 when CPU writes 0 to I2C[3:0]x6C[Enable] while the I²C controller is receiving the address byte of the slave-transmitter operation from a remote master or receiving the address and data bytes of the slave-receiver operation from a remote master. 1=The I²C controller has forced a NACK during any part of an I²C transfer, irrespective of whether the I²C address matches the slave address set in I2C[3:0]x08[IcSar]; Or the transfer is completed before I2C[3:0]x6C[Enable] is set to 0 but not taken effect; Or the remote I²C master terminates the transfer with a STOP condition before the I²C controller can NACK a transfer while I2C[3:0]x6C[Enable] is programmed from 1 to 0. 0=The I²C controller has been disabled when there is master activity, or when the I²C bus is idle.
0	**IcEn**. Read-only. Reset: 0. When I2C[3:0]x6C[Enable] == 1, this bit is forced to 1. When I2C[3:0]x6C[Enable] is programmed from 1 to 0, a delay occurs for this bit to be read as 0 because disabling the I²C controller depends on I²C bus activities. 1=The I²C controller is deemed to be in an enabled state. 0=The I²C controller is deemed to be completely inactive; the CPU can safely read bit SlvRxDataLost and SlvDisabledWhileBusy. This bit can be read at anytime.

I2C[3:0]xA0 I²C SS and FS Spike Suppression Limit Register (IC_FS_SPKLEN)

Bits	Description
31:8	Reserved.
7:0	**IcFsSpkLen**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 5h. Specifies the duration, measured in ic_clk cycles, of the longest spike that is filtered out by the spike suppression logic when the component is operating in Standard Speed (SS) or Full Speed (FS) modes. This register must be set before any I²C bus transaction can take place to ensure stable operation. The minimum valid value is 1; the hardware prevents values less than this being written, and if attempted, results in 1 being set.

I2C[3:0]xA4 I²C HS Spike Suppression Limit Register (IC_HS_SPKLEN)

Bits	Description
31:8	Reserved.
7:0	**IcHsSpkLen**. IF (I2C[3:0]x6C[Enable] == 0) THEN Read-write. ELSE Read-only. ENDIF. Reset: 1. IF (I2C[3:0]xF4[MaxSpeedMode] < 3) THEN Reserved. ELSE Specifies the duration, measured in ic_clk cycles, of the longest spike that is filtered out by the spike suppression logic when the component is operating in High Speed (HS) mode. This register must be set before any I²C bus transaction can take place to ensure stable operation. The minimum valid value is 1; the hardware prevents values less than this being written, and if attempted, results in 1 being set. ENDIF.

I2C[3:0]xF4 Component Parameter 1 Register (IC_COMP_PARAM_1)

This register contains the encoded information about the component parameter settings. This register is valid when I2C[3:0]xF4[AddEncodedParams] == 1.

Bits	Description
31:24	Reserved.
23:16	**TxBufferDepth**. Value: FFh. Specifies the depth of the transmit buffer. The buffer is 9 bits wide; 8 bits for the data, and 1 bit for the read or write command. Bits / Definition / Bits / Definition 00h / Reserved / FEh-02h / <TxBufferDepth+1> bytes 01h / 2 bytes / FFh / 256 bytes
15:8	**RxBufferDepth**. Value: FFh. Specifies the depth of the receive buffer. The buffer is 8 bits wide. Bits / Definition / Bits / Definition 00h / Reserved / FEh-02h / <RxBufferDepth+1> bytes 01h / 2 bytes / FFh / 256 bytes
7	**AddEncodedParams**. Value: 1. 1=Software can read I2C[3:0]xF4 to get the encoded parameters. 0=I2C[3:0]xF4 is reserved.
6	**HasDma**. Value: 0. 1=Include the DMA handshaking interface signals at the top-level IO.
5	**IntrIO**. Value: 1. 0=Each interrupt source has its own output. 1=All interrupt sources are combined into a single output.
4	**HcCountValues**. Value: 0. 1=The count registers (I2C[3:0]x14, I2C[3:0]x18, I2C[3:0]x1C and I2C[3:0]x20) are read-only. 0=The count registers are read-writeable.

3:2	**MaxSpeedMode**. Value: 3. Specifies the maximum I²C mode supported. It controls the reset value of I2C[3:0]x00[Speed]. Bits Definition Bits Definition 00b Reserved 10b Fast speed (400 Kbps or 1000 Kbps) 01b Standard speed (100 Kbps) 11b High speed (3.4 Mbps)
1:0	**ApbDataWidth**. Value: 2. Specifies the width of the APB (Advanced Peripheral Bus) data bus. Bits Definition Bits Definition 00b 8 bits 10b 32 bits 01b 16 bits 11b Reserved

I2C[3:0]xF8 I²C Component Version Register (IC_COMP_VERSION)

Bits	Description
31:0	**IcCompVersion**. Read-only. Reset: 3132_312Ah.

I2C[3:0]xFC I²C Component Type Register (IC_COMP_TYPE)

Bits	Description
31:0	**IcCompType**. Read-only. Reset: 4457_0140h.

3.26.21 UART Registers

There are two UART controllers in this FCH. The UART registers are memory-mapped. UART controller 0 registers range from FEDC_6000h to FEDC_6FFFh. UART controller 1 registers range from FEDC_8000h to FEDC_8FFFh.

UART[1,0]x00 DataBuffer and Divisor Latch Low

IF (UART[1,0]x0C[DLAB] == 0) THEN

UART[1,0]x00 Receive Buffer Register (RBR)

Bits	Description
31:8	Reserved.
7:0	**RBR: Receive Buffer Register**. Read-only. Reset: 0. Specifies the data byte received on the serial input port (sin) in UART mode. The data in this register is valid only if UART[1,0]x14[DR] == 1.

UART[1,0]x00 Transmit Holding Register (THR)

Bits	Description
31:8	Reserved.
7:0	**THR: Transmit Holding Register**. Write-only. Reset: 0. Specifies the data to be transmitted on the serial output port (sout) in UART mode. Data should only be written to this register when UART[1,0]x14[THRE] == 1.

ENDIF.
IF (UART[1,0]x0C[DLAB] == 1) THEN

UART[1,0]x00 Divisor Latch Low (DLL)

Bits	Description
31:8	Reserved.
7:0	**DLL: Divisor Latch Low**. Read-write. Reset: 0. Specifies the lower 8 bits of a 16-bit Divisor Latch register that contains the baud rate divisor for the UART. Baud Rate = (Serial Clock Freq) / (16 * {UART[1,0]x04[DLH], DLL}). • Serial Clock Freq == 48 MHz. If {UART[1,0]x04[DLH], DLL} == 0, the baud clock is disabled.

ENDIF.

IF (UART[1,0]x0C[DLAB] == 1) THEN

UART[1,0]x04 Divisor Latch High (DLH)

Bits	Description
31:8	Reserved.
7:0	**DLH: Divisor Latch High**. Read-write. Reset: 0. Specifies the upper 8 bits of a 16-bit Divisor Latch register that contains the baud rate divisor for the UART. See UART[1,0]x00[DLL].

ENDIF.
IF (UART[1,0]x0C[DLAB] == 0) THEN

UART[1,0]x04 Interrupt Enable Register (IER)

Bits	Description
31:8	Reserved.
7	**PTIME: Programmable THRE Interrupt Mode Enable**. Read-write. Reset: 0. 1=Enable the generation of Transmitter Holding Register Empty (THRE) Interrupt. 0=Disable.
6:4	Reserved.
3	**EDSSI: Enable Modem Status Interrupt**. Read-write. Reset: 0. 1=Enable the generation of Modem Status Interrupt, which is the fourth highest priority interrupt. 0=Disabled.
2	**ELSI: Enable Receiver Line Status Interrupt**. Read-write. Reset: 0. 1=Enable the generation of Receiver Line Status Interrupt, which is the highest priority interrupt. 0=Disabled.
1	**ETBEI: Enable Transmit Holding Register Empty Interrupt**. Read-write. Reset: 0. 1=Enable the generation of Transmitter Holding Register Empty Interrupt, which is the third highest priority interrupt. 0=Disabled.
0	**ERBFI: Enable Received Data Available Interrupt**. Read-write. Reset: 0. 1=Enable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (If in FIFO mode and FIFOs enabled). 0=Disable. These are the second highest priority interrupts.

ENDIF.

IF (READ) THEN

UART[1,0]x08 Interrupt Identification Register (IIR)

Bits	Description
31:8	Reserved.
7:6	**FIFOSE: FIFOs Enabled**. Read-only. Reset: 0. Indicates whether the FIFOs are enabled or disabled. Bits — Definition 00b — Disabled. 10b-01b — Reserved. 11b — Enabled.

5:4	Reserved.
3:0	**IID: Interrupt ID**. Read-only. Reset: 1. Indicates the highest priority pending interrupt which can be one of the following types: Bits — Definition 0000b — Modem status interrupt. Fourth priority. 0001b — Reserved. 0010b — THR empty interrupt. Third priority. 0011b — Reserved. 0100b — Received data available interrupt. Second priority. 0101b — Reserved. 0110b — Receiver line status interrupt. Highest priority. 0111b — Busy detect interrupt. Fifth priority. 1011b-1000b — Reserved. 1100b — Character timeout interrupt. Second priority. 1111b-1101b — Reserved.

ELSE

UART[1,0]x08 FIFO Control Register (FCR)

Bits	Description
31:8	Reserved.
7:6	**RT: RCVR Trigger**. Write-only. Reset: 0. Selects the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. Bits — Definition — Bits — Definition 00b — 1 character in the FIFO — 10b — FIFO 1/2 full 01b — FIFO 1/4 full — 11b — FIFO 2 characters less than full
5:4	**TET: TX Empty Trigger**. Write-only. Reset: 0. Selects the empty threshold level at which the THRE Interrupts are generated when the mode is active. Bits — Definition — Bits — Definition 00b — FIFO empty — 10b — FIFO 1/4 full 01b — 2 characters in the FIFO — 11b — FIFO 1/2 full
3	**DMAM: DMA Mode**. Write-only. Reset: 0. Determines the DMA signalling mode. 0=Mode 0. 1=Mode 1.
2	**XFIFOR: XMIT FIFO Reset**. Write-only; Cleared-by-hardware. Reset: 0. 1=Reset the control portion of the transmit FIFO and treats the FIFO as empty; De-assert the DMA TX request and single signals.
1	**RFIFOR: RCVR FIFO Reset**. Write-only; Cleared-by-hardware. Reset: 0. 1=Reset the control portion of the receive FIFO and treats the FIFO as empty; De-assert the DMA RX request and single signals.
0	**FIFOE: FIFO Enable**. Write-only. Reset: 0. 1=Eable the transmit (XMIT) and receive (RCVR) FIFOs. 0=Disable. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFOs is reset.

ENDIF.

UART[1,0]x0C Line Control Register (LCR)

Bits	Description
31:8	Reserved.
7	**DLAB: Divisor Latch Access Bit**. Read-write. Reset: 0. 1=Enable reading and writing of the Divisor Latch register ({UART[1,0]x04[DLH], UART[1,0]x00[DLL]} to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers.
6	**BC: Break Control**. Read-write. Reset: 0. This is used to cause a break condition to be transmitted to the receiving device. 1=The serial output is forced to the spacing (logic 0) state.
5	**StickParity**. Read-write. Reset: 0. This bit is used to force parity value. 0=Stick Parity is disabled. 1=The parity bit is transmitted and checked as logic 0 if ((PEN == 1) && (EPS == 1)); The parity bit is transmitted and checked as a logic 1 if ((PEN == 1) && (EPS == 0)).
4	**EPS: Even Parity Select**. Read-write. Reset: 0. Selects between even and odd parity when PEN == 1. 1=An even number of logic 1s is transmitted or checked. 0=An odd number of logic 1s is transmitted or checked.
3	**PEN: Parity Enable**. Read-write. Reset: 0. 1=Enable parity generation and detection in transmitted and received serial character respectively. 0=Disable.
2	**STOP**. Read-write. Reset: 0. Selects the number of stop bits per character that the peripheral transmits and receives. 0=One stop bit is transmitted in the serial data. 1=One and a half stop bits is transmitted if (DLS == 0); Otherwise, two stop bits are transmitted.
1:0	**DLS: Data Length Select**. Read-write. Reset: 0. Selects the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected are as follows: Bits Definition Bits Definition 00b 5 bits 10b 7 bits 01b 6 bits 11b 8 bits

UART[1,0]x10 Modem Control Register (MCR)

Bits	Description
31:7	Reserved.
6	**SIRE: SIR Mode Enable**. Value: 0. 0=IrDA SIR Mode disabled.
5	**AFCE: Auto Flow Control Enable**. Read-write. Reset: 0. 0=Auto Flow Control Mode disabled. 1=Auto Flow Control Mode enabled.
4	**LB: LoopBack**. Read-write. Reset: 0. This is used to put the UART into a diagnostic mode for test purposes.
3	**OUT2**. Read-write. Reset: 0. This is used to directly control the user-designated Output2 (out2_n) output. The value written to this location is inverted and driven out on out2_n. 0=out2_n de-asserted (logic 1). 1=out2_n asserted (logic 0).
2	**OUT1**. Read-write. Reset: 0. This is used to directly control the user-designated Output1 (out1_n) output. The value written to this location is inverted and driven out on out1_n. 0=out1_n de-asserted (logic 1). 1=out1_n asserted (logic 0).

1	**RTS: Request to Send**. Read-write. Reset: 0. This is used to directly control the Request to Send (rts_n) output.
0	**DTR: Data Terminal Ready**. Read-write. Reset: 0. This is used to directly control the Data Terminal Ready (dtr_n) output. The value written to this location is inverted and driven out on dtr_n. 0=dtr_n de-asserted (logic 1). 1=dtr_n asserted (logic 0).

UART[1,0]x14 Line Status Register (LSR)

Bits	Description
31:8	Reserved.
7	**RFE: Receiver FIFO Error**. Read-only. Reset: 0. This bit is only valid if UART[1,0]x08[FIFOE] == 1. This specifies if there is at least one parity error, framing error, or break indication in the FIFO. 0=No error in RX FIFO. 1=Error in RX FIFO. This bit is cleared when LSR is read and the character with the error is at the top of the receiver FIFO and there are no subsequent errors in the FIFO.
6	**TEMT: Transmitter Empty**. Read-only. Reset: 1. IF (UART[1,0]x08[FIFOE] == 1) THEN 1=The Transmitter Shift Register and the FIFO are both empty. ELSE 1=The Transmitter Holding Register and the Transmitter Shift Register are both empty. ENDIF.
5	**THRE: Transmit Holding Register Empty**. Read-only. Reset: 1. If UART[1,0]x04[PTIME] == 0, this bit indicates that the THR or TX FIFO is empty which can cause a THRE Interrupt to occur if the THRE Interrupt is enabled. If UART[1,0]x04[PTIME] == 1 && UART[1,0]x08[FIFOE] == 1, the functionality is switched to indicate the transmitter FIFO is full, and no longer controls THRE interrupts, which are then controlled by the UART[1,0]x08[TET] threshold setting.
4	**BI: Break Interrupt**. Read-only. Reset: 0. Indicates the detection of a break sequence on the serial input data. Reading LSR clears this bit.
3	**FE: Framing Error**. Read-only. Reset: 0. Indicates the occurrence of a framing error in the receiver. Reading LSR clears this bit. 0=No framing error. 1=Framing error.
2	**PE: Parity Error**. Read-only. Reset: 0. Indicates the occurrence of a parity error in the receiver if UART[1,0]x0C[PEN] == 1. Reading LSR clears this bit. 0=No parity error. 1=Parity error.
1	**OE: Overrun error**. Read-only. Reset: 0. Indicates the occurrence of an overrun error. Reading LSR clears this bit. 0=No overrun error. 1=Overrun error.
0	**DR: Data Ready**. Read-only; Updated-by-hardware. Reset: 0. Indicates that the receiver contains at least one character in UART[1,0]x00[RBR] or the receiver FIFO. This bit is cleared when the receiver FIFO is empty. 0=No data ready. 1=Data ready.

UART[1,0]x18 Modem Status Register (MSR)

Bits	Description
31:8	Reserved.
7	**DCD: Data Carrier Detect**. Read-only. Reset: 1. Indicates the current state of the modem control line dcd_n. This bit is the complement of dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set. 0=dcd_n input is de-asserted (logic 1). 1=dcd_n input is asserted (logic 0). In Loopback Mode (UART[1,0]x10[LB] == 1), DCD is the same as UART[1,0]x10[OUT2].

6	**RI: Ring Indicator**. Read-only. Reset: 1. Indicates the current state of the modem control line ri_n. This bit is the complement of ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set. 0=ri_n input is de-asserted (logic 1). 1=ri_n input is asserted (logic 0). In Loopback Mode (UART[1,0]x10[LB] == 1), RI is the same as UART[1,0]x10[OUT1].
5	**DSR: Data Set Ready**. Read-only. Reset: 1. Indicates the current state of the modem control line dsr_n. This bit is the complement of dsr_n. When the Data Set Ready input (dsr_n) is asserted it is an indication that the modem or data set is ready to establish communications with the UART controller. 0=dsr_n input is de-asserted (logic 1). 1=dsr_n input is asserted (logic 0). In Loopback Mode (UART[1,0]x10[LB] == 1), DSR is the same as UART[1,0]x10[DTR].
4	**CTS: Clear to Send**. Read-only. Reset: 0. Indicates the current state of the modem control line cts_n. This bit is the complement of cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with the UART controller. 0=cts_n input is de-asserted (logic 1). 1=cts_n input is asserted (logic 0). In Loopback Mode (UART[1,0]x10[LB] == 1), CTS is the same as UART[1,0]x10[RTS].
3	**DDCD: Delta Data Carrier Detect**. Read-only. Reset: 0. 1=The modem control line dcd_n has changed since the last time the MSR was read. 0=No change on dcd_n since last read of MSR. Reading MSR clears this bit. In Loopback Mode (UART[1,0]x10[LB] == 1), this bit reflects changes on UART[1,0]x10[OUT2]. Note, if the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit is set when the reset is removed if the dcd_n signal remains asserted.
2	**TERI: Trailing Edge of Ring Indicator**. Read-only. Reset: 0. 1=A change on the input ri_n (from an active-low to an inactive-high state) has occurred since the last time the MSR was read. 0=No change on ri_n since last read of MSR. Reading MSR clears this bit. In Loopback Mode (UART[1,0]x10[LB] == 1), TERI reflects when UART[1,0]x10[OUT1] has changed state from a high to a low.
1	**DDSR: Delta Data Set Ready**. Read-only. Reset: 0. 1=The modem control line dsr_n has changed since the last time the MSR was read. 0=No change on dsr_n since last read of MSR. Reading MSR clears the DDSR bit. In Loopback Mode (UART[1,0]x10[LB] == 1), DDSR reflects changes on UART[1,0]x10[DTR]. Note, if the DDSR bit is not set and the dsr_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDSR bit is set when the reset is removed if the dsr_n signal remains asserted.
0	**DCTS: Delta Clear to Send**. Read-only. Reset: 0. 1=The modem control line cts_n has changed since the last time the MSR was read. 0=No change on cts_n since last read of MSR. Reading MSR clears the DCTS bit. In Loopback Mode (UART[1,0]x10[LB] == 1), DCTS reflects changes on UART[1,0]x10[RTS]. Note, if the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit is set when the reset is removed if the cts_n signal remains asserted.

UART[1,0]x1C Scratchpad Register (SCR)

Bits	Description
31:8	Reserved.
7:0	**SCR: Scratchpad Register**. Read-write. Reset: 0. This field is for programmers to use as a temporary storage space. It controls no hardware.

IF (READ) THEN

UART[1,0]x[64:30:Step4] Shadow Receive Buffer Register (SRBR)

Bits	Description
31:8	Reserved.
7:0	**SRBR: Shadow Receive Buffer Register**. Read-only. Reset: 0. This is a shadow register for UART[1,0]x00[RBR] and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master.

ELSE

UART[1,0]x[6C:30:Step4] Shadow Transmit Holding Register (STHR)

Bits	Description
31:8	Reserved.
7:0	**STHR: Shadow Transmit Holding Register**. Write-only. Reset: 0. This is a shadow register for UART[1,0]x00[THR] and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master.

ENDIF.

UART[1,0]x70 FIFO Access Register (FAR)

Bits	Description
31:1	Reserved.
0	**FAR: FIFO Access Register**. Read-write. Reset: 0. 1=Enable a FIFO access mode for testing when FIFOs are implemented and enabled; When FIFOs are not implemented or not enabled it allows the RBR to be written by the master and the THR to be read by the master. 0=FIFO access mode disabled.

UART[1,0]x74 Transmit FIFO Read (TFR)

Bits	Description
31:8	Reserved.
7:0	**TFR: Transmit FIFO Read**. Read-only. Reset: 0. These bits are only valid when UART[1,0]x70[FAR] == 1. When FIFOs are implemented and enabled, reading this register gives the data at the top of the transmit FIFO. Each consecutive read pops the transmit FIFO and gives the next data value that is currently at the top of the FIFO. When FIFOs are not implemented or not enabled, reading this register gives the data in UART[1,0]x00[THR].

UART[1,0]x78 Receive FIFO Write (RFW)

Bits	Description
31:10	Reserved.

9	**RFFE: Receive FIFO Framing Error**. Write-only. Reset: 0. These bits are only valid when UART[1,0]x70[FAR] == 1. When FIFOs are implemented and enabled, this bit is used to write framing error detection information to the receive FIFO. When FIFOs are not implemented or not enabled, this bit is used to write framing error detection information to UART[1,0]x00[RBR].
8	**RFPE: Receive FIFO Parity Error**. Write-only. Reset: 0. These bits are only valid when UART[1,0]x70[FAR] == 1. When FIFOs are implemented and enabled, this bit is used to write parity error detection information to the receive FIFO. When FIFOs are not implemented or not enabled, this bit is used to write parity error detection information to UART[1,0]x00[RBR].
7:0	**RFWD: Receive FIFO Write Data**. Write-only. Reset: 0. These bits are only valid when UART[1,0]x70[FAR] == 1. When FIFOs are implemented and enabled, the data that is written to the RFWD is pushed into the receive FIFO. Each consecutive write pushes the new data to the next write location in the receive FIFO. When FIFOs are not implemented or not enabled, the data that is written to the RFWD is pushed into UART[1,0]x00[RBR].

UART[1,0]x7C UART Status Register (USR)

Bits	Description
31:5	Reserved.
4	**RFF: Receive FIFO Full**. Read-only; Updated-by-hardware. Reset: 0. 0=Receive FIFO not full. 1=Receive FIFO Full.
3	**RFNE: Receive FIFO Not Empty**. Read-only; Updated-by-hardware. Reset: 0. 0=Receive FIFO is empty. 1=Receive FIFO is not empty.
2	**TFE: Transmit FIFO Empty**. Read-only; Updated-by-hardware. Reset: 1. 0=Transmit FIFO is not empty. 1=Transmit FIFO is empty.
1	**TFNF: Transmit FIFO Not Full**. Read-only; Updated-by-hardware. Reset: 1. 0=Transmit FIFO is full. 1=Transmit FIFO is not full.
0	**BUSY: UART Busy**. Read-only. Reset: 0. Not used in this implementation.

UART[1,0]x80 Transmit FIFO Level (TFL)

Bits	Description
31:0	**TFL: Transmit FIFO Level**. Read-only; Updated-by-hardware. Reset: 0. Indicates the number of data entries in the transmit FIFO.

UART[1,0]x84 Receive FIFO Level (RFL)

Bits	Description
31:0	**RFL: Receive FIFO Level**. Read-only; Updated-by-hardware. Reset: 0. Indicates the number of data entries in the receive FIFO.

UART[1,0]x88 Software Reset Register (SRR)

Bits	Description
31:3	Reserved.
2	**XFR: XMIT FIFO Reset**. Write-only. Reset: 0. This is a shadow register for UART[1,0]x08[XFIFOR].
1	**RFR: RCVR FIFO Reset**. Write-only; Cleared-by-hardware. Reset: 0. This is a shadow register for UART[1,0]x08[RFIFOR].
0	**UR: UART Reset**. Write-only. Reset: 0. This asynchronously resets UART and synchronously removes the reset assertion. For a two clock implementation both pclk and sclk domains are reset.

UART[1,0]x8C Shadow Request To Send (SRTS)

Bits	Description
31:1	Reserved.
0	**SRTS: Shadow Request to Send**. Read-write. Reset: 0. This is a shadow register for UART[1,0]x10[RTS].

UART[1,0]x90 Shadow Break Control Register (SBCR)

Bits	Description
31:1	Reserved.
0	**SBCR: Shadow Break Control Register**. Read-write. Reset: 0. This is a shadow register for UART[1,0]x0C[BC].

UART[1,0]x94 Shadow DMA Mode (SDMAM)

Bits	Description
31:1	Reserved.
0	**SDMAM: Shadow DMA Mode**. Read-write. Reset: 0. This is a shadow register for UART[1,0]x08[DMAM].

UART[1,0]x98 Shadow FIFO Enable (SFE)

Bits	Description
31:1	Reserved.
0	**SFE: Shadow FIFO Enable**. Read-write. Reset: 0. This is a shadow register for UART[1,0]x08[FIFOE].

UART[1,0]x9C Shadow RCVR Trigger (SRT)

Bits	Description
31:2	Reserved.
1:0	**SRT: Shadow RCVR Trigger**. Read-write. Reset: 0. This is a shadow register for UART[1,0]x08[RT].

UART[1,0]xA0 Shadow TX Empty Trigger (STET)

Bits	Description
31:2	Reserved.
1:0	**STET: Shadow TX Empty Trigger**. Read-write. Reset: 0. This is a shadow register for UART[1,0]x08[TET].

UART[1,0]xA4 Halt TX (HTX)

Bits	Description
31:1	Reserved.
0	**HTX: Halt TX**. Read-write. Reset: 0. 1=Halt transmission is enabled for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled. 0=Halt transmission disabled.

UART[1,0]xA8 DMA Software Acknowledge (DMASA)

Bits	Description
31:1	Reserved.
0	**DMASA: DMA Software Acknowledge**. Write-only; Cleared-by-hardware. Reset: 0. 1=Perform a DMA software acknowledge if a transfer needs to be terminated due to an error condition.

UART[1,0]xF4 Component Parameter Register (CPR)

Bits	Description
31:24	Reserved.
23:16	**FIFO_MODE**. Value: 10h. Specifies Receiver and Transmitter FIFO depth in bytes. 10h=256 bytes. Bits / Definition / Bits / Definition 00h / 0 / 7Fh-03h / <16 * FIFO_MODE> Bytes 01h / 16 Bytes / 80h / 2048 Bytes 02h / 32 Bytes / FFh-81h / Reserved
15:14	Reserved.
13	**DMA_EXTRA**. Value: 1. 1=Additional DMA signals are included.
12	**UART_ADD_ENCODED_PARAMS**.Value: 1. 1=The peripheral has a Component Parameter Register (CPR).

11	**SHADOW**. Value: 1. 1=The peripheral to have seven additional registers that shadow some of the existing register bits that are regularly modified by software.
10	**FIFO_STAT**. Value: 1. 1=The peripheral has three additional FIFO status registers.
9	**FIFO_ACCESS**. Value:1. 1=The peripheral to have a programmable FIFO access mode.
8	**ADDITIONAL_FEAT**. Value: 1. 1=The peripheral supports the FIFO status registers, shadow registers, the encoded parameter register, the UART component version and the peripheral ID registers.
7	**SIR_LP_MODE**. Value: 0. 0=Low-power IrDA SIR mode is not available.
6	**SIR_MODE**. Value: 0. 0=IrDA SIR infrared mode is not available.
5	**THRE_MODE**. Value: 1. 1=The peripheral has a programmable Transmitter Hold Register Empty (THRE) Interrupt mode.
4	**AFCE_MODE**. Value: 1. 1=The peripheral supports the 16750-compatible Auto Flow Control mode.
3:2	Reserved.
1:0	**APB_DATA_WIDTH**. Value: 2. Specifies the width of APB data bus to which this component is attached. Bits Definition Bits Definition 00b 8 bits 10b 32 bits 01b 16 bits 11b Reserved

UART[1,0]xF8 UART Component Version (UCV)

Bits	Description
31:0	**UCV: UART Component Version**. Read-only. Reset: 3331_342Ah. Specifies the ASCII value for each number in the version, followed by *. Version 2.01* is supported.

UART[1,0]xFC Component Type Register (CTR)

Bits	Description
31:0	**PeripheralID**. Read-only. Reset: 4457_0110h. Specifies the peripherals identification code.

UART[1,0]x400 Rx Control Register (RxCtrl)

Bits	Description
31:1	Reserved.
0	**CRxTc: Clear Rx transfer count**. Write-only; Cleared-by-hardware. Reset: 0h. 1=Clear RX transfer counter count.

UART[1,0]x404 Rx Transfer Count Register(RxTransCount)

Bits	Description
31:16	Reserved.
15:0	**RxTc: Rx Transfer Count**. Read-only. Reset: 0h. Specifies the RX transfer count in bytes. The total number of bytes that have been transferred from UART for this RX request. Any time RBR register is read and UART[1,0]x0C[DLAB] == 0 this register is incremented by 1.

UART[1,0]x410 Tx Control Register (TxCtrl)

Bits	Description
31:1	Reserved.
0	**CTxTc: Clear Tx transfer count**. Write-only; Cleared-by-hardware. Reset: 0h. 1=Clear TX transfer counter count.

UART[1,0]x414 Tx Transfer Count Register(TxTransCount)

Bits	Description
31:16	Reserved.
15:0	**TxTc: Tx Transfer Count**. Read-only. Reset: 0. Specifies the TX transfer count in bytes. The total number of bytes that have been transferred to UART for this TX request. Any time THR register is written and UART[1,0]x0C[DLAB] == 0 this register is incremented by 1.

UART[1,0]x420 DmaIrq Register (DmaIrq)

Bits	Description
31:1	Reserved.
0	**DmaIrqEn: DMA Interrupt Request Enable**. Read-write. Reset: 1. 1=Enable DMA Interrupt Request.

4 Register List

The following is a list of all storage elements, context, and registers provided in this document. Page numbers, register mnemonics, and register names are provided.

48 SMMFEC0: SMM IO Trap Offset
49 SMMFEC4: Local SMI Status
49 SMMFEC8: SMM IO Restart Byte
50 SMMFEC9: Auto Halt Restart Offset
50 SMMFECA: NMI Mask
50 SMMFED8: SMM SVM State
51 SMMFEFC: SMM-Revision Identifier
51 SMMFF00: SMM Base Address (SMM_BASE)
191 IOCF8: IO-Space Configuration Address
192 IOCFC: IO-Space Configuration Data Port
193 D0F0x00: Device/Vendor ID
193 D0F0x04: Status/Command
193 D0F0x08: Class Code/Revision ID
193 D0F0x0C: Header Type
194 D0F0x2C: Subsystem and Subvendor ID
194 D0F0x34: Capabilities Pointer
194 D0F0x48: NB Header Write Register
194 D0F0x4C: PCI Control
195 D0F0x60: Miscellaneous Index
195 D0F0x64: Miscellaneous Index Data
195 D0F0x64_x00: Northbridge Control
195 D0F0x64_x0C: IOC Bridge Control
195 D0F0x64_x0D: IOC PCI Configuration
196 D0F0x64_x16: IOC Advanced Error Reporting Control
196 D0F0x64_x17: Memory Mapped IO Base Address
196 D0F0x64_x18: Memory Mapped IO Limit
196 D0F0x64_x19: Top of Memory 2 Low
196 D0F0x64_x1A: Top of Memory 2 High
197 D0F0x64_x1D: Internal Graphics PCI Control
197 D0F0x64_x1F: FCH Location
197 D0F0x64_x22: LCLK Control 0
198 D0F0x64_x23: LCLK Control 1
198 D0F0x64_x3[4:0]: Programmable Device Remap Register
199 D0F0x64_x46: IOC Features Control
199 D0F0x64_x50: ACG Control
200 D0F0x7C: IOC Configuration Control
200 D0F0x84: Link Arbitration
200 D0F0x90: Northbridge Top of Memory
200 D0F0x94: Northbridge ORB Configuration Offset
201 D0F0x98: Northbridge ORB Configuration Data Port
201 D0F0x98_x02: ORB PGMEM Control
201 D0F0x98_x06: ORB Downstream Control 0
202 D0F0x98_x07: ORB Upstream Arbitration Control 0
202 D0F0x98_x08: ORB Upstream Arbitration Control 1
203 D0F0x98_x09: ORB Upstream Arbitration Control 2
203 D0F0x98_x0C: ORB Upstream Arbitration Control 5
203 D0F0x98_x1E: ORB Receive Control 0
204 D0F0x98_x26: ORB IOMMU Control 0
204 D0F0x98_x27: ORB IOMMU Control 1
204 D0F0x98_x28: ORB Transmit Control 0
204 D0F0x98_x29: ORB Transmit Status
205 D0F0x98_x2C: ORB Clock Control
205 D0F0x98_x37: ORB Allow LDTSTOP Control 0
206 D0F0x98_x3A: ORB Source Tag Translation Control 2
206 D0F0x98_x3B: ORB Source Tag Translation Control 3
206 D0F0x98_x4[A,9]: ORB LCLK Clock Control 1-0
207 D0F0xB8: SMU Index Address
207 D0F0xBC: SMU Index Data
207 D0F0xBC_x1300_007C: MP1_C2PMSG_31
207 D0F0xBC_xC001_7150: SVI Loadline Configuration
208 D0F0xBC_xD020_008C: LCLK Deep Sleep Control
208 D0F0xBC_xD021_1058: SPR P-State Power Information 1
208 D0F0xBC_xD021_105C: SPR P-State Power Information 2
209 D0F0xBC_xD021_1060: SPR P-State Power Information 3
209 D0F0xBC_xD021_1074: SPR Product Information Register 1
209 D0F0xBC_xD021_1088: SPR Brand Name Address
209 D0F0xBC_xD021_108C: SPR Brand Name Data Port
210 D0F0xBC_xD021_108C_x[B:0]: SPR Brand Name Data
210 D0F0xBC_xD820_0C64: Hardware Temperature Control (HTC)
211 D0F0xBC_xD820_0CA4: Reported Temperature Control
212 D0F0xBC_xD820_0CE4: Thermtrip Status
213 D0F0xBC_xD822_20B8: COR0_CLK PLL LOCK TIMER
213 D0F0xBC_xD822_2114: COR0_CLK Gater Sequence Register
213 D0F0xBC_xD823_0F00: SMUSVI Defaults
213 D0F0xC8: DEV Index Address
214 D0F0xCC: DEV Index Data
214 D0F0xCC_x01_ib[23:19,15:11]: IOC Bridge Control
215 D0F0xCC_x02_ib[23:19,15:11]: IOC Bridge Status
215 D0F0xD0: GBIF Index Address
215 D0F0xD4: GBIF Index Data
215 D0F0xD4_x0109_14E1: CC Bif Bx Strap0 Ind
216 D0F0xD4_x0109_14E2: CC Bif Bx Strap1 Ind
216 D0F0xD4_x0109_1507: CC Bif Bx Pinstrap0 Ind
216 D0F0xE0: Link Index Address
217 D0F0xE4: Link Index Data
218 D0F0xE4_x0150_0046: Subsystem and Vendor ID
218 D0F0xE4_x0150_0080: Link Configuration
218 D0F0xE4_x0150_0[C:8]00: Link Hold Training Control
218 D0F0xE4_x0150_0[C:8]03: Link Deemphasis Control
219 D0F0xE4_x0130_8002: IO Link Wrapper Scratch
219 D0F0xE4_x0130_80F0: BIOS Timer
219 D0F0xE4_x0130_80F1: BIOS Timer Control
219 D0F0xE4_x0140_0002: IO Link Hardware Debug
219 D0F0xE4_x0140_0010: IO Link Control 1
220 D0F0xE4_x0140_0011: IO Link Config Control
220 D0F0xE4_x0140_001C: IO Link Control 2
221 D0F0xE4_x0140_0020: IO Link Chip Interface Control
221 D0F0xE4_x0140_0040: IO Link Phy Control
221 D0F0xE4_x0140_00B0: IO Link Strap Control
221 D0F0xE4_x0140_00C0: IO Link Strap Miscellaneous
222 D0F0xE4_x0140_00C1: IO Link Strap Miscellaneous2
222 D0F0xF8: Northbridge IOAPIC Index
222 D0F0xFC: Northbridge IOAPIC Data
222 D0F0xFC_x00: IOAPIC Feature Control Register
223 D0F0xFC_x01: IOAPIC Base Address Lower
223 D0F0xFC_x02: IOAPIC Base Address Upper
223 D0F0xFC_x0F: IOAPIC GBIF Interrupt Routing Register
223 D0F0xFC_x1[4:0]: IOAPIC BR Interrupt Routing Register
224 D0F0xFC_x2F: IOAPIC APG Interrupt Routing Register
224 D0F0xFC_x30: IOAPIC SPG Interrupt Routing Register
224 D0F0xFC_x31: IOAPIC Serial IRQ Status
224 D0F0xFC_x3[F:E]: IOAPIC Scratch [1:0] Register
225 D0F2x00: Device/Vendor ID
225 D0F2x04: Status/Command
226 D0F2x08: Class Code/Revision ID
226 D0F2x0C: Header Type
226 D0F2x2C: Subsystem and Subvendor ID
226 D0F2x34: Capabilities Pointer
226 D0F2x3C: Interrupt Line

227 D0F2x40: IOMMU Capability
227 D0F2x44: IOMMU Base Address Low
228 D0F2x48: IOMMU Base Address High
228 D0F2x4C: IOMMU Range
228 D0F2x50: IOMMU Miscellaneous Information Register
229 D0F2x54: IOMMU Miscellaneous Information Register High
229 D0F2x64: IOMMU MSI Capability Register
230 D0F2x68: IOMMU MSI Address Low
230 D0F2x6C: IOMMU MSI Address High
230 D0F2x70: IOMMU MSI Data
230 D0F2x74: IOMMU MSI Mapping Capability
231 D0F2x7C: IOMMU Control
231 D0F2x80: IOMMU MMIO Control Low
232 D0F2x84: IOMMU MMIO Control High
232 D0F2x88: IOMMU Range Control
232 D0F2x8C: IOMMU DSFX Control
232 D0F2x90: IOMMU DSSX Control
233 D0F2x94: IOMMU MARC Status
233 D0F2xF0: IOMMU L2 Config Index
234 D0F2xF4: IOMMU L2 Config Data
234 D0F2xF4_x00: L2_PERF_CNTL_0
234 D0F2xF4_x01: L2_PERF_COUNT_0
234 D0F2xF4_x02: L2_PERF_COUNT_1
234 D0F2xF4_x03: L2_PERF_CNTL_1
234 D0F2xF4_x04: L2_PERF_COUNT_2
235 D0F2xF4_x05: L2_PERF_COUNT_3
235 D0F2xF4_x08: L2_STATUS_0
235 D0F2xF4_x0C: L2_CONTROL_0
236 D0F2xF4_x0D: L2_CONTROL_1
236 D0F2xF4_x10: L2_DTC_CONTROL
237 D0F2xF4_x11: L2_DTC_HASH_CONTROL
237 D0F2xF4_x12: L2_DTC_WAY_CONTROL
237 D0F2xF4_x14: L2_ITC_CONTROL
238 D0F2xF4_x15: L2_ITC_HASH_CONTROL
238 D0F2xF4_x16: L2_ITC_WAY_CONTROL
239 D0F2xF4_x18: L2_PTC_A_CONTROL
239 D0F2xF4_x19: L2_PTC_A_HASH_CONTROL
240 D0F2xF4_x1A: L2_PTC_A_WAY_CONTROL
240 D0F2xF4_x1C: L2_PTC_B_CONTROL
241 D0F2xF4_x1D: L2_PTC_B_HASH_CONTROL
241 D0F2xF4_x1E: L2_PTC_B_WAY_CONTROL
241 D0F2xF4_x20: L2_CREDIT_CONTROL_2
242 D0F2xF4_x22: L2A_UPDATE_FILTER_CNTL
242 D0F2xF4_x30: L2_ERR_RULE_CONTROL_3
242 D0F2xF4_x31: L2_ERR_RULE_CONTROL_4
242 D0F2xF4_x32: L2_ERR_RULE_CONTROL_5
242 D0F2xF4_x33: L2_L2A_CK_GATE_CONTROL
243 D0F2xF4_x34: L2_L2A_PGSIZE_CONTROL
243 D0F2xF4_x35: L2_L2A_PGMEM_CONTROL_1
243 D0F2xF4_x36: L2_L2A_PGMEM_CONTROL_2
244 D0F2xF4_x37: L2_L2A_PGMEM_CONTROL_3
244 D0F2xF4_x38: L2_L2A_PGMEM_CONTROL_4
244 D0F2xF4_x3B: IOMMU_PGFSM_CONFIG
244 D0F2xF4_x3C: IOMMU_PGFSM_WRITE
244 D0F2xF4_x3D: IOMMU_PGFSM_READ
244 D0F2xF4_x3E: L2_PG_CNTL_0
245 D0F2xF4_x3F: L2_PG_CNTL_1
245 D0F2xF4_x40: L2_PG_CNTL_2
245 D0F2xF4_x41: L2_PG_CNTL_3
245 D0F2xF4_x48: L2_STATUS_1
245 D0F2xF4_x49: L2_SB_LOCATION
245 D0F2xF4_x4C: L2_CONTROL_5
246 D0F2xF4_x4D: L2_CONTROL_6
246 D0F2xF4_x50: L2_PDC_CONTROL
247 D0F2xF4_x51: L2_PDC_HASH_CONTROL
248 D0F2xF4_x52: L2_PDC_WAY_CONTROL
248 D0F2xF4_x53: L2B_UPDATE_FILTER_CNTL
248 D0F2xF4_x54: L2_TW_CONTROL
249 D0F2xF4_x56: L2_CP_CONTROL
249 D0F2xF4_x57: L2_CP_CONTROL_1
249 D0F2xF4_x58: IOMMU_L2_GUEST_ADDR_CNTRL
249 D0F2xF4_x6A: L2_INT_CONTROL
250 D0F2xF4_x70: L2_CREDIT_CONTROL_0
250 D0F2xF4_x71: L2_CREDIT_CONTROL_1
250 D0F2xF4_x78: L2_MCIF_CONTROL
251 D0F2xF4_x80: L2_ERR_RULE_CONTROL_0
251 D0F2xF4_x81: L2_ERR_RULE_CONTROL_1
251 D0F2xF4_x82: L2_ERR_RULE_CONTROL_2
251 D0F2xF4_x90: L2_L2B_CK_GATE_CONTROL
252 D0F2xF4_x92: PPR_CONTROL
252 D0F2xF4_x94: L2_L2B_PGSIZE_CONTROL
252 D0F2xF4_x95: L2_L2B_PGMEM_CONTROL_1
253 D0F2xF4_x96: L2_L2B_PGMEM_CONTROL_2
253 D0F2xF4_x97: L2_L2B_PGMEM_CONTROL_3
253 D0F2xF4_x98: L2_L2B_PGMEM_CONTROL_4
253 D0F2xF4_x99: L2_PERF_CNTL_2
253 D0F2xF4_x9A: L2_PERF_COUNT_4
253 D0F2xF4_x9B: L2_PERF_COUNT_5
254 D0F2xF4_x9C: L2_PERF_CNTL_3
254 D0F2xF4_x9D: L2_PERF_COUNT_6
254 D0F2xF4_x9E: L2_PERF_COUNT_7
254 D0F2xF8: IOMMU L1 Config Index
255 D0F2xFC: IOMMU L1 Config Data
255 D0F2xFC_x00_L1i[3:0]: L1_PERF_CNTL
255 D0F2xFC_x01_L1i[3:0]: L1_PERF_COUNT_0
255 D0F2xFC_x02_L1i[3:0]: L1_PERF_COUNT_1
255 D0F2xFC_x07_L1i[3:0]: L1_DEBUG_1
256 D0F2xFC_x09_L1i[3:0]: L1_SB_LOCATION
256 D0F2xFC_x0C_L1i[3:0]: L1_CNTRL_0
257 D0F2xFC_x0D_L1i[3:0]: L1_CNTRL_1
257 D0F2xFC_x0E_L1i[3:0]: L1_CNTRL_2
258 D0F2xFC_x0F_L1i[3:0]: L1_CNTRL_3
258 D0F2xFC_x10_L1i[3:0]: L1_BANK_SEL_0
258 D0F2xFC_x11_L1i[3:0]: L1_BANK_DISABLE_0
258 D0F2xFC_x20_L1i[3:0]: L1_WQ_STATUS_0
259 D0F2xFC_x21_L1i[3:0]: L1_WQ_STATUS_1
259 D0F2xFC_x22_L1i[3:0]: L1_WQ_STATUS_2
260 D0F2xFC_x23_L1i[3:0]: L1_WQ_STATUS_3
260 D0F2xFC_x32_L1i[3:0]: L1_CNTRL_4
261 D0F2xFC_x33_L1i[3:0]: L1_CLKCNTRL_0
261 D0F2xFC_x34_L1i[3:0]: L1_MEMPWRCNTRL_0
261 D0F2xFC_x35_L1i[3:0]: L1_MEMPWRCNTRL_1
261 D0F2xFC_x36_L1i[3:0]: L1_GUEST_ADDR_CNTRL
262 D0F2xFC_x37_L1i[3:0]: L1_FEATURE_SUP_CNTRL
262 D0F2xFC_x38_L1i[3:0]: L1_CNTRL_5
262 D0F2xFC_x39_L1i[3:0]: L1_PGMEM_CNTRL1
263 D0F2xFC_x3A_L1i[3:0]: L1_PGMEM_CNTRL2
263 D0F2xFC_x3B_L1i[3:0]: L1_PGMEM_CNTRL3
263 D0F2xFC_x3C_L1i[3:0]: L1_PGMEM_CNTRL4
263 D0F2xFC_x3D_L1i[3:0]: L1_SST_CNTRL0
263 D0F2xFC_x3E_L1i[3:0]: L1_ATS_RESP_CNTRL0
264 D1F0x00: Device/Vendor ID
264 D1F0x04: Status/Command Register
265 D1F0x08: Class Code/Revision ID Register
265 D1F0x0C: Header Type Register
265 D1F0x10: Graphic Memory Base Address

266 D1F0x14: Graphics Memory Base Address 64
266 D1F0x18: Graphics Doorbell Base Address
266 D1F0x1C: Graphics Doorbell Base Address 64
266 D1F0x20: Graphics IO Base Address
267 D1F0x24: Graphics Memory Mapped Registers Base Address
267 D1F0x2C: Subsystem and Subvendor ID Register
267 D1F0x30: Expansion ROM Base Address
267 D1F0x34: Capabilities Pointer
267 D1F0x3C: Interrupt Line
268 D1F0x4C: Subsystem and Subvendor ID Mirror
268 D1F0x50: Power Management Capability
268 D1F0x54: Power Management Control and Status
269 D1F0x58: PCI Express® Capability
269 D1F0x5C: Device Capability
269 D1F0x60: Device Control and Status
270 D1F0x64: Link Capability
271 D1F0x68: Link Control and Status
272 D1F0x7C: Device Capability 2
272 D1F0x80: Device Control and Status 2
273 D1F0x84: Link Capability 2
273 D1F0x88: Link Control and Status 2
274 D1F0xA0: MSI Capability
274 D1F0xA4: MSI Message Address Low
274 D1F0xA8: MSI Message Address High
274 D1F0xAC: MSI Message Data
275 D1F0x100: Vendor Specific Enhanced Capability
275 D1F0x104: Vendor Specific Header
275 D1F0x108: Vendor Specific 1
275 D1F0x10C: Vendor Specific 2
276 D1F1x00: Device/Vendor ID
276 D1F1x04: Status/Command
277 D1F1x08: Class Code/Revision ID
277 D1F1x0C: Header Type
277 D1F1x10: Audio Registers Base Address
277 D1F1x14: Base Address 1
278 D1F1x18: Base Address 2
278 D1F1x1C: Base Address 3
278 D1F1x20: Base Address 4
278 D1F1x24: Base Address 5
278 D1F1x2C: Subsystem and Subvendor ID
278 D1F1x30: Expansion ROM Base Address
279 D1F1x34: Capabilities Pointer
279 D1F1x3C: Interrupt Line
279 D1F1x4C: Subsystem and Subvendor ID Mirror
279 D1F1x50: Power Management Capability
280 D1F1x54: Power Management Control and Status
280 D1F1x58: PCI Express® Capability
280 D1F1x5C: Device Capability
281 D1F1x60: Device Control and Status
282 D1F1x64: Link Capability
282 D1F1x68: Link Control and Status
283 D1F1x7C: Device Capability 2
283 D1F1x80: Device Control and Status 2
284 D1F1x84: Link Capability 2
284 D1F1x88: Link Control and Status 2
284 D1F1xA0: MSI Capability
285 D1F1xA4: MSI Message Address Low
285 D1F1xA8: MSI Message Address High
285 D1F1xAC: MSI Message Data
285 D1F1x100: Vendor Specific Enhanced Capability
285 D1F1x104: Vendor Specific Header
286 D1F1x108: Vendor Specific 1
286 D1F1x10C: Vendor Specific 2
287 D2F0x00: Device/Vendor ID (Host Bridge)
287 D2F0x04: Status/Command
287 D2F0x08: Class Code/Revision ID
287 D2F0x0C: Header Type
288 D2F0x40: Header Type Write
288 D2F[5:1]x00: Device/Vendor ID
288 D2F[5:1]x04: Status/Command Register
289 D2F[5:1]x08: Class Code/Revision ID Register
289 D2F[5:1]x0C: Header Type Register
290 D2F[5:1]x18: Bus Number and Secondary Latency Register
290 D2F[5:1]x1C: IO Base and Secondary Status Register
291 D2F[5:1]x20: Memory Limit and Base Register
291 D2F[5:1]x24: Prefetchable Memory Limit and Base Register
291 D2F[5:1]x28: Prefetchable Memory Base High Register
291 D2F[5:1]x2C: Prefetchable Memory Limit High Register
291 D2F[5:1]x30: IO Base and Limit High Register
292 D2F[5:1]x34: Capabilities Pointer Register
292 D2F[5:1]x3C: Bridge Control Register
292 D2F[5:1]x50: Power Management Capability Register
293 D2F[5:1]x54: Power Management Control and Status Register
293 D2F[5:1]x58: PCI Express® Capability Register
294 D2F[5:1]x5C: Device Capability Register
294 D2F[5:1]x60: Device Control and Status Register
295 D2F[5:1]x64: IO Link Capability Register
296 D2F[5:1]x68: IO Link Control and Status Register
298 D2F[5:1]x6C: Slot Capability Register
299 D2F[5:1]x70: Slot Control and Status Register
300 D2F[5:1]x74: Root Complex Capability and Control Register
300 D2F[5:1]x78: Root Complex Status Register
300 D2F[5:1]x7C: Device Capability 2
301 D2F[5:1]x80: Device Control and Status 2
302 D2F[5:1]x84: IO Link Capability 2
302 D2F[5:1]x88: IO Link Control and Status 2
303 D2F[5:1]x8C: Slot Capability 2
304 D2F[5:1]x90: Slot Control and Status 2
304 D2F[5:1]xA0: MSI Capability Register
304 D2F[5:1]xA4: MSI Message Address Low
304 D2F[5:1]xA8: MSI Message Address High
305 D2F[5:1]xAC: MSI Message Data
305 D2F[5:1]xC0: Subsystem and Subvendor Capability ID Register
305 D2F[5:1]xC4: Subsystem and Subvendor ID Register
305 D2F[5:1]xC8: MSI Mapping Capability
305 D2F[5:1]xCC: MSI Mapping Address Low
306 D2F[5:1]xD0: MSI Mapping Address High
306 D2F[5:1]xE0: Root Port Index
306 D2F[5:1]xE4: Root Port Data
306 D2F[5:1]xE4_x20: Root Port TX Control
306 D2F[5:1]xE4_x50: Root Port Lane Status
307 D2F[5:1]xE4_x6A: Root Port Error Control
307 D2F[5:1]xE4_x70: Root Port Receiver Control
307 D2F[5:1]xE4_xA0: Per Port Link Controller (LC) Control
308 D2F[5:1]xE4_xA1: LC Training Control
308 D2F[5:1]xE4_xA2: LC Link Width Control
309 D2F[5:1]xE4_xA3: LC Number of FTS Control
310 D2F[5:1]xE4_xA4: LC Link Speed Control
310 D2F[5:1]xE4_xA5: LC State 0
311 D2F[5:1]xE4_xB1: LC Control 2
311 D2F[5:1]xE4_xB5: LC Control 3
312 D2F[5:1]xE4_xC0: LC Strap Override
312 D2F[5:1]xE4_xC1: Root Port Miscellaneous Strap Override
312 D2F[5:1]xE4_xD0: Root Port ECC Skip OS Feature
313 D2F[5:1]xE4_xD2: PCIEP_HPGI_PRIVATE
313 D2F[5:1]xE4_xDA: PCIEP_HPGI

314 D2F[5:1]x100: Vendor Specific Enhanced Capability Register
314 D2F[5:1]x104: Vendor Specific Header Register
314 D2F[5:1]x108: Vendor Specific 1 Register
314 D2F[5:1]x10C: Vendor Specific 2 Register
314 D2F[5:1]x128: Virtual Channel 0 Resource Status Register
315 D2F[5:1]x136: Virtual Channel 1 Resource Status Register
315 D2F[5:1]x150: Advanced Error Reporting Capability
315 D2F[5:1]x154: Uncorrectable Error Status
316 D2F[5:1]x158: Uncorrectable Error Mask
317 D2F[5:1]x15C: Uncorrectable Error Severity
317 D2F[5:1]x160: Correctable Error Status
318 D2F[5:1]x164: Correctable Error Mask
318 D2F[5:1]x168: Advanced Error Control
319 D2F[5:1]x16C: Header Log DW0
319 D2F[5:1]x170: Header Log DW1
319 D2F[5:1]x174: Header Log DW2
319 D2F[5:1]x178: Header Log DW3
319 D2F[5:1]x17C: Root Error Command
320 D2F[5:1]x180: Root Error Status
320 D2F[5:1]x184: Error Source ID
321 D9F2x00: Device/Vendor ID
321 D9F2x04: Status/Command
322 D9F2x08: Class Code/Revision ID
322 D9F2x0C: Header Type
322 D9F2x10: BAR0 Address Map
322 D9F2x14: BAR1 Address Map
322 D9F2x24: MSI-X BAR Address Map
323 D9F2x34: Capabilities Pointer
323 D9F2x3C: Interrupt Line
323 D9F2x40: ACGAZ Mirror Reg Ctrl 0
323 D9F2x44: ACGAZ Mirror Reg Ctrl 1
324 D9F2x50: ACGAZ MSI-X Capability Register
324 D9F2x54: ACGAZ MSI-X Table Structure Register
325 D9F2x58: ACGAZ PBA Structure Register
325 D9F2x5C: ACGAZ HyperTransport™ MSI Capability Register
325 D9F2x60: ACGAZ PM Capability Register
326 D9F2x68: ACGAZ PCIe® Capability Register
326 D9F2x6C: ACGAZ PCIe® Device Capability
327 D9F2x70: ACGAZ PCIe® Device Ctrl Status
327 D9F2x8C: ACGAZ PCIe® Device Capability 2
328 D9F2x90: ACGAZ PCIe® Device Ctrl Status 2
328 D9F2xA4: ACGAZ PCI Advance Features Capability Register
328 D9F2xA8: ACGAZ PCI Advance Features Ctrl Status
329 D9F2xE8: NB ACG CMN Config Index(NB ACG CMN INDEX)
329 D9F2xEC: NB ACG CMN Config Data(NB ACG CMN DATA)
329 D9F2xEC_x4A: ACG LCLK Clock Gating Control0
330 D9F2xEC_x4C: ACG AXICLK Clock Gating Control0
331 D9F2xF4_x0F: ACGAZ Downstream BAR Mapping
331 D9F2xF4_x1[3:0]: ACGAZ Downstream BAR
331 D9F2xF4_x20: ACGAZ Upstream Control
332 D9F2xF4_x21: ACGAZ Upstream PASID Control
332 D9F2xF4_x2[9:8]: ACGAZ Upstream Aper0 BAR
333 D9F2xF4_x2[B:A]: ACGAZ Upstream Aper0 Limit Register
333 D9F2xF4_x2[D:C]: ACGAZ Upstream Aper0 Map Register
334 D9F2xF4_x48: ACGAZ Interrupt Polarity Control Register
334 D9F2xF4_x49: ACGAZ Interrupt Enable Register
335 D18F0x00: Device/Vendor ID
335 D18F0x04: Status/Command
335 D18F0x08: Class Code/Revision ID
335 D18F0x0C: Header Type
335 D18F0x34: Capabilities Pointer
336 D18F0x[5C:40]: Routing Table
336 D18F0x60: Node ID
336 D18F0x64: Unit ID
337 D18F0x68: Link Transaction Control
338 D18F0x6C: Link Initialization Control
339 D18F0x[E4,C4,A4,84]: Link Control
340 D18F0x[EC,CC,AC,8C]: Link Feature Capability
340 D18F0x[F0,D0,B0,90]: Link Base Channel Buffer Count
343 D18F0x[F4,D4,B4,94]: Link Isochronous Channel Buffer Count
343 D18F0x[F8,D8,B8,98]: Link Type
344 D18F0x[11C,118,114,110]: Link Clumping Enable
344 D18F0x150: Link Global Retry Control
344 D18F0x168: Extended Link Transaction Control
344 D18F0x16C: Link Global Extended Control
345 D18F0x[18C:170]: Link Extended Control
345 D18F0x1A0: Link Initialization Status
345 D18F0x1DC: Core Enable
346 D18F0x1F8: ONION3 Link Status and Control Register
346 D18F0x20[4:0]: ONION3 Upstream Base Channel Buffer Count
348 D18F1x00: Device/Vendor ID
348 D18F1x08: Class Code/Revision ID
348 D18F1x0C: Header Type
348 D18F1x[17C:140,7C:40]: DRAM Base/Limit
350 D18F1x[2CC:2A0,1CC:180,BC:80]: MMIO Base/Limit
354 D18F1x[DC:C0]: IO-Space Base/Limit
356 D18F1x[1DC:1D0,EC:E0]: Configuration Map
356 D18F1xF0: DRAM Hole Address
357 D18F1xF4: VGA Enable
358 D18F1x10C: DCT Configuration Select
358 D18F1x120: DRAM Base System Address
358 D18F1x124: DRAM Limit System Address
359 D18F1x2[1C:00]: DRAM Controller Base/Limit
361 D18F2x00: Device/Vendor ID
361 D18F2x08: Class Code/Revision ID
361 D18F2x0C: Header Type
361 D18F2x[5C:40]_dct[0]: DRAM CS Base Address
363 D18F2x[6C:60]_dct[0]: DRAM CS Mask
363 D18F2x78_dct[0]: DRAM Control
364 D18F2x7C_dct[0]: DRAM Initialization
365 D18F2x80_dct[0]: DRAM Bank Address Mapping
367 D18F2x84_dct[0]: DRAM MRS
368 D18F2x88_dct[0]: DRAM Timing Low
368 D18F2x8C_dct[0]: DRAM Timing High
368 D18F2x90_dct[0]: DRAM Configuration Low
370 D18F2x94_dct[0]: DRAM Configuration High
373 D18F2x98_dct[0]: DRAM Controller Additional Data Offset
374 D18F2x9C_dct[0]: DRAM Controller Additional Data Port
374 D18F2x9C_x00[F,3:0]0_0009_dct[0]: High Addr Mode
374 D18F2x9C_x0000_000E_dct[0]: Global Control Slave
374 D18F2x9C_x00[F,3:0]0_0013_dct[0]: NB Pstate
375 D18F2x9C_x0[3,1:0][F,3:0]0_0014_dct[0]: Dll Lock Maintenance
375 D18F2x9C_x00F0_0015_dct[0]: Vref Byte
375 D18F2x9C_x00[F,3:0]0_001A_dct[0]: ByteDbgCtrl
376 D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]028_dct[0]: ABYTE RdPtrOffset
376 D18F2x9C_x00[F,3:0]0_[F,2:0][8,3:0]2E_dct[0]: ABYTE RdPtrInitVal
376 D18F2x9C_x0[3,1:0][F,3:0]0_[F,B:0]041_dct[0]: ABYTE Tx Impedance
377 D18F2x9C_x00[F,3:0]0_[F,B:0]04A_dct[0]: ABYTE DqDqsRcvCntrl1
378 D18F2x9C_x0[F,1:0][F,3:0]0_[F,B:0]04E_dct[0]: TxControlDq
378 D18F2x9C_x00[F,3:0]0_[F,B:0]050_dct[0]: TxControlDq2
379 D18F2x9C_x00[F,3:0]0_[F,B:0]05F_dct[0]: ABYTE Tx Slew Rate
379 D18F2x9C_x0[3,1:0][F,3:0]0_006D_dct[0]: CkPhaseCtl
380 D18F2x9C_x00[F,3:0]0_0077_dct[0]: DllPowerdown
380 D18F2x9C_x00[F,3:0]0_0078_dct[0]: DllControl
380 D18F2x9C_x0[3,1:0][F,3:0]0_[F,2:0]081_dct[0]: ABYTE TxDly

381 D18F2x9C_x00[F,8:0]1_0000_dct[0]: VariousChicken
381 D18F2x9C_x00[F,8:0]1_000E_dct[0]: Global Control Slave
382 D18F2x9C_x0[3,1:0][F,8:0]1_0010_dct[0]: DByteCtrl1
382 D18F2x9C_x00[F,8:0]1_0013_dct[0]: NB Pstate
383 D18F2x9C_x0[3,1:0][F,8:0]1_0014_dct[0]: Dll Lock Maintenance
383 D18F2x9C_x00[F,8:0]1_0015_dct[0]: Vref Byte
384 D18F2x9C_x00[F,8:0]1_0016_dct[0]: Proc Odt Timing
385 D18F2x9C_x00[F,8:0]1_001A_dct[0]: ByteDbgCtrl
385 D18F2x9C_x00[F,8:0]1_001C_dct[0]: DynPwrDnUp
385 D18F2x9C_x0[3,1:0][F,8:0]1_0028_dct[0]: DATA RdPtrOffset
386 D18F2x9C_x0[3,1:0][F,8:0]1_0029_dct[0]: EarlyTrafficOffset
386 D18F2x9C_x0[3,1:0][F,8:0]1_002A_dct[0]: RxStaggerCnt
387 D18F2x9C_x0[3,1:0][F,8:0]1_002B_dct[0]: Tx Dll Standby Stagger Config
387 D18F2x9C_x0[3,1:0][F,8:0]1_002C_dct[0]: Rx Pad Traffic Early Offset
388 D18F2x9C_x00[F,8:0]1_0[8,3:0]2E_dct[0]: DATA RdPtrInitVal
388 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]041_dct[0]: TxImpedanceDq
389 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]043_dct[0]: DqDqsRcvCntrl2
390 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]044_dct[0]: VrefHspeed
390 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]045_dct[0]: VrefLpower
391 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]046_dct[0]: TxStrenHi
391 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]047_dct[0]: TxStrenLo
392 D18F2x9C_x00[F,8:0]1_[F,B:0]04A_dct[0]: DqDqsRcvCntrl1
392 D18F2x9C_x0[3,1:0][F,8:0]1_[F,B:0]04D_dct[0]: DATA Rx Impedance
393 D18F2x9C_x00[F,8:0]1_[F,B:0]04E_dct[0]: TxControlDq
394 D18F2x9C_x00[F,8:0]1_[F,B:0]050_dct[0]: TxControlDq2
394 D18F2x9C_x00[F,8:0]1_[F,B:0]051_dct[0]: DqDqsRcvCntrl3
395 D18F2x9C_x00[F,8:0]1_[F,B:0]05F_dct[0]: DATA Tx Slew Rate
395 D18F2x9C_x00[F,8:0]1_0[F,2:0]77_dct[0]: DllPowerdown
396 D18F2x9C_x00[F,8:0]1_0[F,2:0]78_dct[0]: DllControl
396 D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]80_dct[0]: Rx Delay
397 D18F2x9C_x0[3,1:0][F,8:0]1_[F,3:0][F,3:0]81_dct[0]: Tx Delay
397 D18F2x9C_x0002_0000_dct[0]: PLL MemoryPstate0
398 D18F2x9C_x0002_0001_dct[0]: PLL MemoryPstate1
398 D18F2x9C_x0002_0004_dct[0]: Mailbox Protocol Shadow
398 D18F2x9C_x0002_000B_dct[0]: Power State Command
399 D18F2x9C_x0002_000E_dct[0]: Global Control
399 D18F2x9C_x0002_0015_dct[0]: Vref Byte
399 D18F2x9C_x0002_001A_dct[0]: ByteDbgCtrl
399 D18F2x9C_x0002_0032_dct[0]: Upstream Mailbox 1 Message
399 D18F2x9C_x0002_0033_dct[0]: Upstream Mailbox 1 Protocol
400 D18F2x9C_x0002_0034_dct[0]: Upstream Mailbox 2 Message
400 D18F2x9C_x0002_0035_dct[0]: Upstream Mailbox 2 Protocol
400 D18F2x9C_x0002_005A_dct[0]: D3_MERR_RCVR_CNTRL
400 D18F2x9C_x0002_005B_dct[0]: D3_EVNT_RCVR_CNTRL
401 D18F2x9C_x0002_005D_dct[0]: Misc5
401 D18F2x9C_x0002_005F_dct[0]: Misc PHY Status
401 D18F2x9C_x0002_0060_dct[0]: Memreset Control
401 D18F2x9C_x0[3,1:0]02_0080_dct[0]: PMU CLK Divider
402 D18F2x9C_x0002_0087_dct[0]: CalAndPllConfig
403 D18F2x9C_x0002_0088_dct[0]: CalRate
403 D18F2x9C_x0002_0089_dct[0]: PllLockTime
403 D18F2x9C_x0002_008A_dct[0]: PllDllLockStatus
403 D18F2x9C_x0002_008B_dct[0]: Pstate
404 D18F2x9C_x0002_008E_dct[0]: MERR_status
404 D18F2x9C_x0002_0093_dct[0]: PllRegWaitTime
404 D18F2x9C_x0002_0097_dct[0]: CalBusy
404 D18F2x9C_x0002_0098_dct[0]: CalMisc2
404 D18F2x9C_x0002_0099_dct[0]: PMU Reset
405 D18F2x9C_x0002_009B_dct[0]: CalVRefs
405 D18F2x9C_x0002_00B2_dct[0]: VrefInGlobal
405 D18F2x[B,0]9C_x0005_[5FFF:4000]_dct[0]: PMU IC SRAM
405 D18F2x9C_x0005_[0FFF:0000]_dct[0]: PMU SRAM Message Block
406 D18F2x9C_x00[F,1:0]4_00E[7:0]_dct[0]: OdtCtrl
406 D18F2x9C_x00[F,1:0]4_00E8_dct[0]: OdtCtrl8
406 D18F2x9C_x00[F,1:0]4_00FD_dct[0]: PHY CKE control
407 D18F2x9C_x0007_0015_dct[0]: Lane to CRC Map0
407 D18F2x9C_x0007_0016_dct[0]: Lane to CRC Map1
408 D18F2xA4: DRAM Controller Temperature Throttle
409 D18F2xA8_dct[0]: DRAM Controller Miscellaneous 2
410 D18F2xAC: DRAM Controller Temperature Status
410 D18F2x110: DRAM Controller Select Low
411 D18F2x114: DRAM Controller Select High
412 D18F2x118: Memory Controller Configuration Low
413 D18F2x11C: Memory Controller Configuration High
415 D18F2x1[7C:40]_dct[0]: DQ Mapping
416 D18F2x1A8_dct[0]: PSM Index
416 D18F2x1AC_dct[0]: PSM Index
416 D18F2x1B0: Extended Memory Controller Configuration Low
418 D18F2x1B4: Extended Memory Controller Configuration High Register
420 D18F2x1B8_dct[0]: DRAM ZQ to CS Map
421 D18F2x1BC_dct[0]: DRAM CKE to CS Map
421 D18F2x1CC_dct[0]: Data Scramble Key
421 D18F2x1[E8,E0,D8,D0]_dct[0]: Performance Event Select Low
422 D18F2x1[EC,E4,DC,D4]_dct[0]: Performance Event Select High
422 D18F2x1F[C:0]_dct[0]: DRAM NB P-State Configuration
423 D18F2x200_dct[0]_mp[1:0]: DDR3 DRAM Timing 0
423 D18F2x204_dct[0]_mp[1:0]: DDR3 DRAM Timing 1
425 D18F2x208_dct[0]_mp[1:0]: DDR3 DRAM Timing 2
425 D18F2x20C_dct[0]_mp[1:0]: DDR3 DRAM Timing 3
426 D18F2x214_dct[0]_mp[1:0]: DDR3 DRAM Timing 4
427 D18F2x218_dct[0]_mp[1:0]: DDR3 DRAM Timing 5
428 D18F2x21C_dct[0]_mp[1:0]: DDR3 DRAM Timing 6
429 D18F2x220_dct[0]: DDR3 DRAM Timing 7
429 D18F2x224_dct[0]: DDR3 DRAM Timing 8
430 D18F2x228_dct[0]: DDR3 DRAM Timing 9
430 D18F2x22C_dct[0]_mp[1:0]: DDR3 DRAM Timing 10
430 D18F2x[234:230]_dct[0]: DDR3 DRAM Read ODT Pattern [High:Low]
431 D18F2x[23C:238]_dct[0]: DDR3 DRAM Write ODT Pattern [High:Low]
432 D18F2x240_dct[0]_mp[1:0]: DDR3 DRAM ODT Control
432 D18F2x244_dct[0]: DRAM Controller Miscellaneous 3
433 D18F2x248_dct[0]_mp[1:0]: DRAM Power Management 0
433 D18F2x24C_dct[0]: DDR3 DRAM Power Management 1
434 D18F2x250_dct[0]: DRAM Loopback and Training Control
436 D18F2x25[8,4]_dct[0]: DRAM Target [B, A] Base
436 D18F2x25C_dct[0]: DRAM Command 0
437 D18F2x260_dct[0]: DRAM Command 1
437 D18F2x264_dct[0]: DRAM Status 0
438 D18F2x268_dct[0]: DRAM Status 1
438 D18F2x26C_dct[0]: DRAM Status 2
438 D18F2x270_dct[0]: DRAM PRBS
439 D18F2x274_dct[0]: DRAM DQ Mask Low
439 D18F2x278_dct[0]: DRAM DQ Mask High
439 D18F2x280_dct[0]: DRAM DQ Pattern Override 0
439 D18F2x284_dct[0]: DRAM DQ Pattern Override 1
440 D18F2x288_dct[0]: DRAM DQ Pattern Override 2
440 D18F2x28C_dct[0]: DRAM Command 2
441 D18F2x290_dct[0]: DRAM Status 3
441 D18F2x294_dct[0]: DRAM Status 4
441 D18F2x298_dct[0]: DRAM Status 5
441 D18F2x2A0_dct[0]_mp[1:0]: DRAM Timing 12
442 D18F2x2A4_dct[0]_mp[1:0]: DRAM Timing 13
442 D18F2x2[B4,B0,AC,A8]_dct[0]: DRAM User Data Pattern

443 D18F2x2B8_dct[0]: DRAM Command 3
443 D18F2x2[C0,BC]_dct[0]: DRAM Command 4 & 5
444 D18F2x2E0_dct[0]: Memory P-state Control and Status
445 D18F2x2[F4:E8]_dct[0]_mp[1:0]: MRS Buffer
445 D18F2x2FC_dct[0]: DRAM Timing 11
446 D18F3x00: Device/Vendor ID
446 D18F3x04: Status/Command
446 D18F3x08: Class Code/Revision ID
446 D18F3x0C: Header Type
446 D18F3x34: Capability Pointer
447 D18F3x40: MCA NB Control
449 D18F3x44: MCA NB Configuration
452 D18F3x48: MCA NB Status Low
452 D18F3x4C: MCA NB Status High
458 D18F3x50: MCA NB Address Low
459 D18F3x54: MCA NB Address High
462 D18F3x64: Hardware Thermal Control (HTC)
462 D18F3x68: Software P-state Limit
462 D18F3x6C: Data Buffer Count
463 D18F3x70: SRI to XBAR Command Buffer Count
464 D18F3x74: XBAR to SRI Command Buffer Count
465 D18F3x78: MCT to XBAR Buffer Count
466 D18F3x7C: Free List Buffer Count
467 D18F3x[84:80]: ACPI Power State Control
470 D18F3x88: NB Configuration 1 Low (NB_CFG1_LO)
470 D18F3x8C: NB Configuration 1 High (NB_CFG1_HI)
471 D18F3xA0: Power Control Miscellaneous
471 D18F3xA8: Pop Up and Down P-states
471 D18F3xB8: NB Array Address
472 D18F3xBC: NB Array Data Port
472 D18F3xC0: COFVID Control
472 D18F3xC4: SBI P-state Limit
473 D18F3xC8: COFVID Status Low
473 D18F3xCC: COFVID Status High
474 D18F3xD4: Clock Power/Timing Control 0
475 D18F3xD8: Clock Power/Timing Control 1
475 D18F3xDC: Clock Power/Timing Control 2
478 D18F3xE8: Northbridge Capabilities
478 D18F3xF0: DEV Capability Header Register
479 D18F3xF4: DEV Function Register
479 D18F3xF8: DEV Data Port
479 D18F3xF8_x4: DEV Secure Loader Control Register
479 D18F3xFC: CPUID Family/Model/Stepping
479 D18F3x140: SRI to XCS Token Count
480 D18F3x144: MCT to XCS Token Count
481 D18F3x1[54,50,4C,48]: Link to XCS Token Count
481 D18F3x160: NB Machine Check Misc (DRAM Thresholding) 0 (MC4_MISC0)
482 D18F3x168: NB Machine Check Misc (Link Thresholding) 1 (MC4_MISC1)
482 D18F3x170: NB Machine Check Misc (L3 Thresholding) 2 (MC4_MISC2)
482 D18F3x17C: Extended Freelist Buffer Count
483 D18F3x180: Extended NB MCA Configuration
484 D18F3x188: NB Configuration 2
485 D18F3x190: Downcore Control
485 D18F3x198: MCA NB Control Mask (MC4_CTL_MASK)
486 D18F3x1A0: Core Interface Buffer Count
487 D18F3x1CC: IBS Control
487 D18F3x200: Performance Mode Control Register
488 D18F3x240: MCT Configuration Low (MCT_CFG2_LO)
489 D18F3x244: MCT Configuration High (MCT_CFG2_High)
490 D18F3x2A0: Core Interface Buffer Count Register 2
491 D18F3x2B4: DCT and Fuse Power Gate Control
492 D18F3x2E0: ONION3 Error Status
492 D18F3x2E4: ONION3 Error Address Low
492 D18F3x2E8: ONION3 Error Address High
493 D18F4x00: Device/Vendor ID
493 D18F4x04: Status/Command
493 D18F4x08: Class Code/Revision ID
493 D18F4x0C: Header Type
493 D18F4x34: Capabilities Pointer
494 D18F4x110: Sample and Residency Timers
494 D18F4x11[C:8]: C-state Control
497 D18F4x124: C-state Interrupt Control
497 D18F4x128: C-state Policy Control 1
498 D18F4x13C: SMU P-state Control
499 D18F4x15C: Core Performance Boost Control
499 D18F4x16C: APM TDP Control
500 D18F4x170: Interrupt Rate Monitor Control
501 D18F4x1C0: Node Cac Register 1
501 D18F4x250: TDP Limit 8
502 D18F4x2[D4:D0]: Power Management Override Register 1,0
503 D18F5x00: Device/Vendor ID
503 D18F5x04: Status/Command
503 D18F5x08: Class Code/Revision ID
503 D18F5x0C: Header Type
503 D18F5x34: Capabilities Pointer
503 D18F5x[70,60,50,40]: Northbridge Performance Event Select Low
504 D18F5x[74,64,54,44]: Northbridge Performance Event Select High
504 D18F5x[78,68,58,48]: Northbridge Performance Event Counter Low
504 D18F5x[7C,6C,5C,4C]: Northbridge Performance Event Counter High
504 D18F5x80: Compute Unit Status 1
505 D18F5x84: Northbridge Capabilities 2
505 D18F5x88: NB Configuration 4 (NB_CFG4)
506 D18F5x8C: NB Configuration 5 (NB_CFG5)
506 D18F5xA0: Northbridge Configuration 8(NB_CFG8)
506 D18F5xE0: Processor TDP Running Average
506 D18F5xE8: TDP Limit 3
507 D18F5xEC: Load Step Throttle Control
507 D18F5x128: Clock Power/Timing Control 3
508 D18F5x16[C:0]: Northbridge P-state [3:0]
510 D18F5x170: Northbridge P-state Control
511 D18F5x174: Northbridge P-state Status
512 D18F5x178: Northbridge FCH Configuration
513 D18F5x1[FC:C0]: BIOS Scratch
513 D18F5x21[8:4]: ONION3 Link Controller Buffer Count
514 D18F5x224: Extended MCT Buffer Count
514 D18F5x228: MCT to ONION3 Buffer Count
515 D18F5x22C: ONION3 Configuration
516 GMMx2024: MC_VM_FB_LOCATION
516 GMMx2068: MC_VM_FB_OFFSET
516 GMMx224[C:4]: MC_VM_MB_L1_DEBUG
517 NBIOAPICx00: IO Register Select
517 NBIOAPICx10: IO Window
517 NBIOAPICx10_x00: IOAPIC ID
517 NBIOAPICx10_x01: IOAPIC Version
517 NBIOAPICx10_x02: IOAPIC Arbitration
518 NBIOAPICx10_x[4E:10:step2]: Redirection Table Entry [31:0]
518 NBIOAPICx20: IRQ Pin Assertion
518 NBIOAPICx40: EOI
519 IOMMUx00: Device Table Base Address Low
519 IOMMUx04: Device Table Base Address High
519 IOMMUx08: Command Buffer Base Address Low
519 IOMMUx0C: Command Buffer Base Address High
520 IOMMUx10: Event Log Base Address Low

520 IOMMUx14: Event Log Base Address High
520 IOMMUx18: Control Low
522 IOMMUx1C: Control High
523 IOMMUx20: Exclusion Range Base Low
523 IOMMUx24: Exclusion Range Base High
524 IOMMUx28: Exclusion Range Limit Low
524 IOMMUx2C: Exclusion Range Limit High
524 IOMMUx30: Extended Feature Low
526 IOMMUx34: Extended Feature High
527 IOMMUx38: PPR Log Base Address Low
527 IOMMUx3C: PPR Log Base Address High
527 IOMMUx40: Hardware Error Upper Low
527 IOMMUx44: Hardware Error Upper High
527 IOMMUx48: Hardware Error Lower Low
528 IOMMUx4C: Hardware Error Lower High
528 IOMMUx50: Hardware Error Status
528 IOMMUx[78,70,68,60]: SMI Filter Low
528 IOMMUx[7C,74,6C,64]: SMI Filter High
529 IOMMUxE0: Guest APIC Log Base Address Low
529 IOMMUxE4: Guest APIC Log Base Address High
529 IOMMUxE8: Guest APIC Log Tail Address Low
529 IOMMUxEC: Guest APIC Log Tail Address High
529 IOMMUxF0: PPR LogB Base Address Low
530 IOMMUxF4: PPR LogB Base Address High
530 IOMMUxF8: Event LogB Base Address Low
530 IOMMUxFC: Event LogB Base Address High
530 IOMMUx[130,128,120,118,110,108,100]: Device Table Base Address Low
531 IOMMUx[134,12C,124,11C,114,10C,104]: Device Table Base Address High
531 IOMMUx138: DSFX
531 IOMMUx140: DSCX
532 IOMMUx148: DSSX
532 IOMMUx150: Miscellenous Capabilities Low
532 IOMMUx154: Miscellenous Capabilities High
532 IOMMUx158: MSI Capabilities
532 IOMMUx15C: MSI Address Low
533 IOMMUx160: MSI Address High
533 IOMMUx164: MSI Data
533 IOMMUx168: MSI Mapping Capability
533 IOMMUx16C: MSI Control
533 IOMMUx[248,230,218,200]: MARC Base Address Low
534 IOMMUx[24C,234,21C,204]: MARC Base Address High
534 IOMMUx[250,238,220,208]: MARC Relocation Address Low
535 IOMMUx[254,23C,224,20C]: MARC Relocatin Address High
535 IOMMUx[258,240,228,210]: MARC Length Low
535 IOMMUx[25C,244,22C,214]: MARC Length High
536 IOMMUx1FF8: P2P Data
536 IOMMUx2000: Command Buffer Head Pointer
536 IOMMUx2008: Command Buffer Tail Pointer
536 IOMMUx2010: Event Log Head Pointer
537 IOMMUx2018: Event Log Tail Pointer
537 IOMMUx2020: Status
538 IOMMUx2030: PPR Log Head Pointer
539 IOMMUx2038: PPR Log Tail Pointer
539 IOMMUx2040: Guest APIC Log Head Pointe Low
539 IOMMUx2048: Guest APIC Log Tail Pointer Low
539 IOMMUx2050: PPR LogB Head Pointer
539 IOMMUx2058: PPR LogB Tail Pointer
540 IOMMUx2070: Event LogB Head Pointer
540 IOMMUx2078: Event LogB Tail Pointer
540 IOMMUx2080: PPR Auto Response
541 IOMMUx2088: PPR Log Overflow Early
541 IOMMUx2090: PPR LogB Overflow Early
541 IOMMUx4000: Counter Configuration
542 IOMMUx4008: Counter PASID Bank Lock Low
542 IOMMUx400C: Counter PASID Bank Lock High
542 IOMMUx4010: Domain Bank Lock Low
542 IOMMUx4014: Domain Bank Lock High
542 IOMMUx4018: DeviceID Bank Lock Low
542 IOMMUx401C: DeviceID Bank Lock High
543 IOMMUx4[1,0][3:0]00: Counter Low
543 IOMMUx4[1,0][3:0]04: Counter High
543 IOMMUx4[1,0][3:0]08: Counter Source
544 IOMMUx4[1,0][3:0]10: PASID Match Low
544 IOMMUx4[1,0][3:0]14: PASID Match High
545 IOMMUx4[1,0][3:0]18: Domain Match Low
545 IOMMUx4[1,0][3:0]1C: Domain Match High
545 IOMMUx4[1,0][3:0]20: DeviceID Match Low
546 IOMMUx4[1,0][3:0]24: DeviceID Match High
546 IOMMUx4[1,0][3:0]28: Counter Report Low
547 IOMMUx4[1,0][3:0]2C: Counter Report High
548 APIC20: APIC ID
548 APIC30: APIC Version
548 APIC80: Task Priority (TPR)
548 APIC90: Arbitration Priority (APR)
549 APICA0: Processor Priority (PPR)
549 APICB0: End of Interrupt
549 APICC0: Remote Read
549 APICD0: Logical Destination (LDR)
549 APICE0: Destination Format
550 APICF0: Spurious-Interrupt Vector (SVR)
550 APIC[170:100]: In-Service (ISR)
550 APIC[1F0:180]: Trigger Mode (TMR)
551 APIC[270:200]: Interrupt Request (IRR)
551 APIC280: Error Status
552 APIC300: Interrupt Command Low (ICR Low)
553 APIC310: Interrupt Command High (ICR High)
553 APIC320: LVT Timer
554 APIC330: LVT Thermal Sensor
554 APIC340: LVT Performance Monitor
554 APIC3[60:50]: LVT LINT[1:0]
555 APIC370: LVT Error
555 APIC380: Timer Initial Count
555 APIC390: Timer Current Count
556 APIC3E0: Timer Divide Configuration
556 APIC400: Extended APIC Feature
556 APIC410: Extended APIC Control
557 APIC420: Specific End Of Interrupt
557 APIC[4F0:480]: Interrupt Enable
558 APIC[530:500]: Extended Interrupt [3:0] Local Vector Table
558 CPUID Fn0000_0000_EAX: Processor Vendor and Largest Standard Function Number
559 CPUID Fn0000_0000_E[D,C,B]X: Processor Vendor
559 CPUID Fn0000_0001_EAX: Family, Model, Stepping Identifiers
559 CPUID Fn0000_0001_EBX: LocalApicId, LogicalProcessorCount, CLFlush
560 CPUID Fn0000_0001_ECX: Feature Identifiers
560 CPUID Fn0000_0001_EDX: Feature Identifiers
561 CPUID Fn0000_000[4:2]: Reserved
562 CPUID Fn0000_0005_EAX: Monitor/MWait
562 CPUID Fn0000_0005_EBX: Monitor/MWait
562 CPUID Fn0000_0005_ECX: Monitor/MWait
562 CPUID Fn0000_0005_EDX: Monitor/MWait
562 CPUID Fn0000_0006_EAX: Thermal and Power Management
562 CPUID Fn0000_0006_EBX: Thermal and Power Management

563 CPUID Fn0000_0006_ECX: Thermal and Power Management
563 CPUID Fn0000_0006_EDX: Thermal and Power Management
563 CPUID Fn0000_0007_EAX_x0: Structured Extended Feature Identifiers (ECX=0)
563 CPUID Fn0000_0007_EBX_x0: Structured Extended Feature Identifiers (ECX=0)
563 CPUID Fn0000_0007_ECX_x0: Structured Extended Feature Identifiers (ECX=0)
564 CPUID Fn0000_0007_EDX_x0: Structured Extended Feature Identifiers (ECX=0)
564 CPUID Fn0000_000[A:8]: Reserved
564 CPUID Fn0000_000B: Reserved
564 CPUID Fn0000_000C: Reserved
564 CPUID Fn0000_000D_EAX_x0: Processor Extended State Enumeration (ECX=0)
564 CPUID Fn0000_000D_EBX_x0: Processor Extended State Enumeration (ECX=0)
565 CPUID Fn0000_000D_ECX_x0: Processor Extended State Enumeration (ECX=0)
565 CPUID Fn0000_000D_EDX_x0: Processor Extended State Enumeration (ECX=0)
565 CPUID Fn0000_000D_EAX_x1: Processor Extended State Enumeration (ECX=1)
565 CPUID Fn0000_000D_E[D,C,B]X_x1: Processor Extended State Enumeration (ECX=1)
565 CPUID Fn0000_000D_EAX_x2: Processor Extended State Enumeration (ECX=2)
565 CPUID Fn0000_000D_EBX_x2: Processor Extended State Enumeration (ECX=2)
565 CPUID Fn0000_000D_ECX_x2: Processor Extended State Enumeration (ECX=2)
566 CPUID Fn0000_000D_EDX_x2: Processor Extended State Enumeration (ECX=2)
566 CPUID Fn0000_000D_EAX_x3E: Processor Extended State Enumeration (ECX=62)
566 CPUID Fn0000_000D_EBX_x3E: Processor Extended State Enumeration (ECX=62)
566 CPUID Fn0000_000D_ECX_x3E: Processor Extended State Enumeration (ECX=62)
566 CPUID Fn0000_000D_EDX_x3E: Processor Extended State Enumeration (ECX=62)
566 CPUID Fn8000_0000_EAX: Largest Extended Function Number
567 CPUID Fn8000_0000_E[D,C,B]X: Processor Vendor
567 CPUID Fn8000_0001_EAX: Family, Model, Stepping Identifiers
567 CPUID Fn8000_0001_EBX: BrandId Identifier
568 CPUID Fn8000_0001_ECX: Feature Identifiers
569 CPUID Fn8000_0001_EDX: Feature Identifiers
570 CPUID Fn8000_000[4:2]_E[D,C,B,A]X: Processor Name String Identifier
570 CPUID Fn8000_0005_EAX: L1 TLB 2M/4M Identifiers
571 CPUID Fn8000_0005_EBX: L1 TLB 4K Identifiers
571 CPUID Fn8000_0005_ECX: L1 Data Cache Identifiers
571 CPUID Fn8000_0005_EDX: L1 Instruction Cache Identifiers
572 CPUID Fn8000_0006_EAX: L2 TLB 2M/4M Identifiers
572 CPUID Fn8000_0006_EBX: L2 TLB 4K Identifiers
572 CPUID Fn8000_0006_ECX: L2 Cache Identifiers
573 CPUID Fn8000_0006_EDX: L3 Cache Identifiers
573 CPUID Fn8000_0007_EAX: Processor Feedback Capabilities
573 CPUID Fn8000_0007_EBX: RAS Capabilities
574 CPUID Fn8000_0007_ECX: Advanced Power Management Information
574 CPUID Fn8000_0007_EDX: Advanced Power Management Information
574 CPUID Fn8000_0008_EAX: Long Mode Address Size Identifiers
575 CPUID Fn8000_0008_EBX: Reserved
575 CPUID Fn8000_0008_ECX: Size Identifiers
575 CPUID Fn8000_0008_EDX: Reserved
575 CPUID Fn8000_0009: Reserved
576 CPUID Fn8000_000A_EAX: SVM Revision and Feature Identification
576 CPUID Fn8000_000A_EBX: SVM Revision and Feature Identification
576 CPUID Fn8000_000A_ECX: SVM Revision and Feature Identification
576 CPUID Fn8000_000A_EDX: SVM Revision and Feature Identification
577 CPUID Fn8000_00[18:0B]: Reserved
577 CPUID Fn8000_0019_EAX: L1 TLB 1G Identifiers
577 CPUID Fn8000_0019_EBX: L2 TLB 1G Identifiers
577 CPUID Fn8000_0019_E[D,C]X: Reserved
577 CPUID Fn8000_001A_EAX: Performance Optimization Identifiers
578 CPUID Fn8000_001A_E[D,C,B]X: Reserved
578 CPUID Fn8000_001B_EAX: Instruction Based Sampling Identifiers
578 CPUID Fn8000_001B_E[D,C,B]X: Instruction Based Sampling Identifiers
579 CPUID Fn8000_001C_EAX: Lightweight Profiling Capabilities 0
579 CPUID Fn8000_001C_EBX: Lightweight Profiling Capabilities 0
580 CPUID Fn8000_001C_ECX: Lightweight Profiling Capabilities 0
580 CPUID Fn8000_001C_EDX: Lightweight Profiling Capabilities 0
581 CPUID Fn8000_001D_EAX_x0: Cache Properties
581 CPUID Fn8000_001D_EAX_x1: Cache Properties
582 CPUID Fn8000_001D_EAX_x2: Cache Properties
582 CPUID Fn8000_001D_EAX_x3: Cache Properties
582 CPUID Fn8000_001D_EBX_x0: Cache Properties
583 CPUID Fn8000_001D_EBX_x1: Cache Properties
583 CPUID Fn8000_001D_EBX_x2: Cache Properties
583 CPUID Fn8000_001D_EBX_x3: Cache Properties
584 CPUID Fn8000_001D_ECX_x0: Cache Properties
584 CPUID Fn8000_001D_ECX_x1: Cache Properties
584 CPUID Fn8000_001D_ECX_x2: Cache Properties
584 CPUID Fn8000_001D_ECX_x3: Cache Properties
585 CPUID Fn8000_001D_EDX_x0: Cache Properties
585 CPUID Fn8000_001D_EDX_x1: Cache Properties
585 CPUID Fn8000_001D_EDX_x2: Cache Properties
585 CPUID Fn8000_001D_EDX_x3: Cache Properties
586 CPUID Fn8000_001E_EAX: Extended APIC ID
586 CPUID Fn8000_001E_EBX: Compute Unit Identifiers
586 CPUID Fn8000_001E_ECX: Node Identifiers
586 CPUID Fn8000_001E_EDX: Reserved
587 MSR0000_0000: Load-Store MCA Address
587 MSR0000_0001: Load-Store MCA Status
587 MSR0000_0010: Time Stamp Counter (TSC)
587 MSR0000_001B: APIC Base Address (APIC_BAR)
588 MSR0000_002A: Cluster ID (EBL_CR_POWERON)
588 MSR0000_00E7: Max Performance Frequency Clock Count (MPERF)
588 MSR0000_00E8: Actual Performance Frequency Clock Count (APERF)
588 MSR0000_00FE: MTRR Capabilities (MTRRcap)
588 MSR0000_0174: SYSENTER CS (SYSENTER_CS)
589 MSR0000_0175: SYSENTER ESP (SYSENTER_ESP)
589 MSR0000_0176: SYSENTER EIP (SYSENTER_EIP)
589 MSR0000_0179: Global Machine Check Capabilities (MCG_CAP)
589 MSR0000_017A: Global Machine Check Status (MCG_STAT)
590 MSR0000_017B: Global Machine Check Exception Reporting Control (MCG_CTL)
590 MSR0000_01D9: Debug Control (DBG_CTL_MSR)
590 MSR0000_01DB: Last Branch From IP (BR_FROM)
590 MSR0000_01DC: Last Branch To IP (BR_TO)
591 MSR0000_01DD: Last Exception From IP
591 MSR0000_01DE: Last Exception To IP
591 MSR0000_020[F:0]: Variable-Size MTRRs Base/Mask
593 MSR0000_02[6F:68,59:58,50]: Fixed-Size MTRRs
595 MSR0000_0277: Page Attribute Table (PAT)

595 MSR0000_02FF: MTRR Default Memory Type (MTRRdefType)
596 MSR0000_0400: MC0 Machine Check Control (MC0_CTL)
596 MSR0000_0401: MC0 Machine Check Status (MC0_STATUS)
599 MSR0000_0402: MC0 Machine Check Address (MC0_ADDR)
600 MSR0000_0403: MC0 Machine Check Miscellaneous (MC0_MISC)
601 MSR0000_0404: MC1 Machine Check Control (MC1_CTL)
602 MSR0000_0405: MC1 Machine Check Status (MC1_STATUS)
606 MSR0000_0406: MC1 Machine Check Address (MC1_ADDR)
608 MSR0000_0407: MC1 Machine Check Miscellaneous (MC1_MISC)
609 MSR0000_0408: MC2 Machine Check Control (MC2_CTL)
610 MSR0000_0409: MC2 Machine Check Status (MC2_STATUS)
614 MSR0000_040A: MC2 Machine Check Address (MC2_ADDR)
615 MSR0000_040B: MC2 Machine Check Miscellaneous (MC2_MISC)
616 MSR0000_040C: MC3 Machine Check Control (MC3_CTL)
616 MSR0000_040D: MC3 Machine Check Status (MC3_STATUS)
616 MSR0000_040E: MC3 Machine Check Address (MC3_ADDR)
616 MSR0000_040F: MC3 Machine Check Miscellaneous (MC3_MISC)
616 MSR0000_0410: MC4 Machine Check Control (MC4_CTL)
617 MSR0000_0411: MC4 Machine Check Status (MC4_STATUS)
620 MSR0000_0412: MC4 Machine Check Address (MC4_ADDR)
621 MSR0000_0413: NB Machine Check Misc 4 (MC4_MISC0)
621 MSR0000_0414: MC5 Machine Check Control (MC5_CTL)
622 MSR0000_0415: MC5 Machine Check Status (MC5_STATUS)
624 MSR0000_0416: MC5 Machine Check Address (MC5_ADDR)
625 MSR0000_0417: MC5 Machine Check Miscellaneous (MC5_MISC)
625 MSR0000_0418: MC6 Machine Check Control (MC6_CTL)
626 MSR0000_0419: MC6 Machine Check Status (MC6_STATUS)
627 MSR0000_041A: MC6 Machine Check Address (MC6_ADDR)
627 MSR0000_041B: MC6 Machine Check Miscellaneous (MC6_MISC)
628 MSRC000_0080: Extended Feature Enable (EFER)
628 MSRC000_0081: SYSCALL Target Address (STAR)
628 MSRC000_0082: Long Mode SYSCALL Target Address (STAR64)
629 MSRC000_0083: Compatibility Mode SYSCALL Target Address (STARCOMPAT)
629 MSRC000_0084: SYSCALL Flag Mask (SYSCALL_FLAG_MASK)
629 MSRC000_00E7: Read-Only Max Performance Frequency Clock Count (MPerfReadOnly)
629 MSRC000_00E8: Read-Only Actual Performance Frequency Clock Count (APerfReadOnly)
630 MSRC000_0100: FS Base (FS_BASE)
630 MSRC000_0101: GS Base (GS_BASE)
630 MSRC000_0102: Kernel GS Base (KernelGSbase)
630 MSRC000_0103: Auxiliary Time Stamp Counter (TSC_AUX)
630 MSRC000_0104: Time Stamp Counter Ratio (TscRateMsr)
631 MSRC000_0105: Lightweight Profile Configuration (LWP_CFG)
632 MSRC000_0106: Lightweight Profile Control Block Address (LWP_CBADDR)
632 MSRC000_0408: NB Machine Check Misc 4 (Link Thresholding) 1 (MC4_MISC1)
633 MSRC000_0409: NB Machine Check Misc 4 (L3 Thresholding) 1 (MC4_MISC2)
634 MSRC000_040[F:A]: Reserved
634 MSRC000_0410: Machine Check Deferred Error Configuration (CU_DEFER_ERR)
635 MSRC001_00[03:00]: Performance Event Select (PERF_CTL[3:0])
635 MSRC001_00[07:04]: Performance Event Counter (PERF_CTR[3:0])
635 MSRC001_0010: System Configuration (SYS_CFG)
636 MSRC001_0015: Hardware Configuration (HWCR)
638 MSRC001_00[18,16]: IO Range Base (IORR_BASE[1:0])
639 MSRC001_00[19,17]: IO Range Mask (IORR_MASK[1:0])
639 MSRC001_001A: Top Of Memory (TOP_MEM)
639 MSRC001_001D: Top Of Memory 2 (TOM2)
639 MSRC001_001F: Northbridge Configuration 1 (NB_CFG1)
640 MSRC001_0022: Machine Check Exception Redirection
640 MSRC001_00[35:30]: Processor Name String
641 MSRC001_003E: Hardware Thermal Control (HTC)
641 MSRC001_0044: DC Machine Check Control Mask (MC0_CTL_MASK)
641 MSRC001_0045: IC Machine Check Control Mask (MC1_CTL_MASK)
642 MSRC001_0046: BU Machine Check Control Mask (MC2_CTL_MASK)
643 MSRC001_0047: Reserved (MC3_CTL_MASK)
643 MSRC001_0048: NB Machine Check Control Mask (MC4_CTL_MASK)
643 MSRC001_0049: EX Machine Check Control Mask (MC5_CTL_MASK)
643 MSRC001_004A: FP Machine Check Control Mask (MC6_CTL_MASK)
643 MSRC001_00[53:50]: IO Trap (SMI_ON_IO_TRAP_[3:0])
644 MSRC001_0054: IO Trap Control (SMI_ON_IO_TRAP_CTL_STS)
645 MSRC001_0055: Interrupt Pending
646 MSRC001_0056: SMI Trigger IO Cycle
646 MSRC001_0058: MMIO Configuration Base Address
647 MSRC001_0060: BIST Results
647 MSRC001_0061: P-state Current Limit
647 MSRC001_0062: P-state Control
648 MSRC001_0063: P-state Status
648 MSRC001_00[6B:64]: P-state [7:0]
649 MSRC001_0070: COFVID Control
650 MSRC001_0071: COFVID Status
650 MSRC001_0073: C-state Base Address
650 MSRC001_0074: CPU Watchdog Timer (CpuWdtCfg)
651 MSRC001_007A: Compute Unit Power Accumulator
651 MSRC001_007B: Max Compute Unit Power Accumulator
651 MSRC001_0111: SMM Base Address (SMM_BASE)
652 MSRC001_0112: SMM TSeg Base Address (SMMAddr)
652 MSRC001_0113: SMM TSeg Mask (SMMMask)
653 MSRC001_0114: Virtual Machine Control (VM_CR)
654 MSRC001_0115: IGNNE
654 MSRC001_0116: SMM Control (SMM_CTL)
655 MSRC001_0117: Virtual Machine Host Save Physical Address (VM_HSAVE_PA)
655 MSRC001_0118: SVM Lock Key
655 MSRC001_011A: Local SMI Status
655 MSRC001_011B: AVIC Doorbell
656 MSRC001_0140: OS Visible Work-around MSR0 (OSVW_ID_Length)
656 MSRC001_0141: OS Visible Work-around MSR1 (OSVW Status)
656 MSRC001_020[A,8,6,4,2,0]: Performance Event Select (PERF_CTL[5:0])
658 MSRC001_020[B,9,7,5,3,1]: Performance Event Counter (PERF_CTR[5:0])
658 MSRC001_024[6,4,2,0]: Northbridge Performance Event Select (NB_PERF_CTL[3:0])
659 MSRC001_024[7,5,3,1]: Northbridge Performance Event Counter (NB_PERF_CTR[3:0])
659 MSRC001_0280: Performance Time Stamp Counter (CU_PTSC)
660 MSRC001_1002: CPUID Features for CPUID Fn0000_0007_E[B,A]X_x0
660 MSRC001_1003: Thermal and Power Management CPUID Features
660 MSRC001_1004: CPUID Features (Features)
662 MSRC001_1005: Extended CPUID Features (ExtFeatures)
664 MSRC001_101[B:9]: Address Mask For DR[3:1] Breakpoints
664 MSRC001_101C: Load-Store Configuration 3 (LS_CFG3)
665 MSRC001_1020: Load-Store Configuration (LS_CFG)
665 MSRC001_1021: Instruction Cache Configuration (IC_CFG)

665 MSRC001_1022: Data Cache Configuration (DC_CFG)
665 MSRC001_1023: Combined Unit Configuration (CU_CFG)
666 MSRC001_1027: Address Mask For DR0 Breakpoints (DR0_ADDR_MASK)
666 MSRC001_1028: Floating Point Configuration (FP_CFG)
666 MSRC001_102A: Combined Unit Configuration 2 (CU_CFG2)
668 MSRC001_102B: Combined Unit Configuration 3 (CU_CFG3)
669 MSRC001_102F: Prefetch Throttling Configuration (CU_PFTCFG)
670 MSRC001_1030: IBS Fetch Control (IbsFetchCtl)
671 MSRC001_1031: IBS Fetch Linear Address (IbsFetchLinAd)
672 MSRC001_1032: IBS Fetch Physical Address (IbsFetchPhysAd)
672 MSRC001_1033: IBS Execution Control (IbsOpCtl)
673 MSRC001_1034: IBS Op Logical Address (IbsOpRip)
673 MSRC001_1035: IBS Op Data (IbsOpData)
674 MSRC001_1036: IBS Op Data 2 (IbsOpData2)
674 MSRC001_1037: IBS Op Data 3 (IbsOpData3)
676 MSRC001_1038: IBS DC Linear Address (IbsDcLinAd)
676 MSRC001_1039: IBS DC Physical Address (IbsDcPhysAd)
676 MSRC001_103A: IBS Control
676 MSRC001_103B: IBS Branch Target Address (BP_IBSTGT_RIP)
677 MSRC001_103C: IBS Fetch Control Extended (IC_IBS_EXTD_CTL)
677 MSRC001_103D: IBS Op Data 4 (DC_IBS_DATA2)
677 MSRC001_1090: Processor Feedback Constants 0
677 MSRC001_10A1: Contention Blocking Buffer Control (CU_CBBCFG)
679 PMCx000: FPU Pipe Assignment
679 PMCx001: FP Scheduler Empty
679 PMCx003: Retired SSE/AVX Operations
680 PMCx004: Number of Move Elimination and Scalar Op Optimization
680 PMCx005: Retired Serializing Ops
680 PMCx020: Segment Register Loads
681 PMCx021: Pipeline Restart Due to Self-Modifying Code
681 PMCx022: Pipeline Restart Due to Various Events
681 PMCx023: Load Queue/Store Queue Full
681 PMCx024: Locked Operations
681 PMCx026: Retired CLFLUSH Instructions
681 PMCx027: Retired CPUID Instructions
682 PMCx029: LS Dispatch
682 PMCx02A: Canceled Store to Load Forward Operations
682 PMCx02B: SMIs Received
682 PMCx030: Executed CLFLUSH Instructions
682 PMCx032: Misaligned Stores
682 PMCx034: FP +Load Buffer Stall
682 PMCx035: STLF
682 PMCx040: Data Cache Accesses
683 PMCx041: Data Cache Misses
683 PMCx042: Data Cache Refills from L2 or System
683 PMCx043: Data Cache Refills from System
683 PMCx045: Unified TLB Hit
684 PMCx046: Unified TLB Miss
684 PMCx047: Misaligned Accesses
684 PMCx04B: Prefetch Instructions Dispatched
684 PMCx052: Ineffective Software Prefetchs
685 PMCx060: Command Related to Victim Buffers
685 PMCx061: Command Related to Masked Operations
685 PMCx062: Command Related to Read Block Operations
686 PMCx063: Command Related to Change to Dirty Operations
686 PMCx064: Dram System Request
686 PMCx065: Memory Requests by Type
686 PMCx067: Data Cache Prefetches
686 PMCx068: MAB Requests
687 PMCx069: MAB Wait Cycles
687 PMCx06C: System Response by Coherence State
687 PMCx06D: Octwords Written to System
688 PMCx075: Cache Cross-invalidates
688 PMCx076: CPU Clocks not Halted
688 PMCx07D: Requests to L2 Cache
688 PMCx07E: L2 Cache Misses
689 PMCx07F: L2 Fill/Writeback
689 PMCx165: Page Splintering
689 PMCx16C: L2 Prefetcher Trigger Events
690 PMCx177: XAB Allocation Stall
690 PMCx17F: Number of free XAB entries available to thread
690 PMCx080: Instruction Cache Fetches
690 PMCx081: Instruction Cache Misses
690 PMCx082: Instruction Cache Refills from L2
690 PMCx083: Instruction Cache Refills from System
691 PMCx084: L1 ITLB Miss, L2 ITLB Hit
691 PMCx085: L1 ITLB Miss, L2 ITLB Miss
691 PMCx086: Pipeline Restart Due to Instruction Stream Probe
691 PMCx087: Instruction Fetch Stall
691 PMCx088: Return Stack Hits
691 PMCx089: Return Stack Overflows
692 PMCx08B: Instruction Cache Victims
692 PMCx08C: Instruction Cache Lines Invalidated
692 PMCx099: ITLB Reloads
692 PMCx09A: ITLB Reloads Aborted
692 PMCx186: Uops Dispatched From Decoder
692 PMCx0C0: Retired Instructions
693 PMCx0C1: Retired uops
693 PMCx0C2: Retired Branch Instructions
693 PMCx0C3: Retired Mispredicted Branch Instructions
693 PMCx0C4: Retired Taken Branch Instructions
693 PMCx0C5: Retired Taken Branch Instructions Mispredicted
693 PMCx0C6: Retired Far Control Transfers
693 PMCx0C7: Retired Branch Resyncs
693 PMCx0C8: Retired Near Returns
693 PMCx0C9: Retired Near Returns Mispredicted
693 PMCx0CA: Retired Mispredicted Taken Branch Instructions due to Target Mismatch
694 PMCx0CB: Retired MMX™/FP Instructions
694 PMCx0CD: Interrupts-Masked Cycles
694 PMCx0CE: Interrupts-Masked Cycles with Interrupt Pending
694 PMCx0CF: Interrupts Taken
694 PMCx0D0: Decoder Empty
694 PMCx0D1: Dispatch Stalls
695 PMCx0D3: Microsequencer Stall due to Serialization
695 PMCx0D5: Dispatch Stall for Instruction Retire Queue Full
695 PMCx0D6: Dispatch Stall for Integer Scheduler Queue Full
695 PMCx0D7: Dispatch Stall for FP Scheduler Queue Full
695 PMCx0D8: Dispatch Stall for LDQ Full
695 PMCx0D9: Microsequencer Stall Waiting for All Quiet
695 PMCx0DB: FPU Exceptions
696 PMCx0D[F:C]: DR[3:0] Breakpoint Matches
696 PMCx1C0: Retired x87 Floating Point Operations
696 PMCx1CF: Tagged IBS Ops
696 PMCx1D0: Retired Fused Branch Instructions
697 PMCx1D8: Dispatch Stall for STQ Full
697 PMCx1DD: Cycles Without Dispatch Due To Integer PRF Tokens
697 PMCx1DE: Cycles Without Dispatch Due to FP PRF Tokens
697 PMCx1DF: FP Dispatch Contention
698 NBPMCx0E4: Memory Controller Bypass Counter Saturation
698 NBPMCx0E8: Thermal Status
698 NBPMCx0E9: CPU or IO Requests to Memory or IO
699 NBPMCx0EA: Cache Block Commands
699 NBPMCx0EB: Sized Commands
700 NBPMCx0EC: Probe Responses and Upstream Requests

701 NBPMCx1E0: CPU to DRAM Requests to Target Node
701 NBPMCx1E1: IO to DRAM Requests to Target Node
701 NBPMCx1E2: CPU Read Command Latency to Target Node 0-3
702 NBPMCx1E3: CPU Read Command Requests to Target Node 0-3
702 NBPMCx1E4: CPU Read Command Latency to Target Node 4-7
703 NBPMCx1E5: CPU Read Command Requests to Target Node 4-7
703 NBPMCx1E6: CPU Command Latency to Target Node 0-3/4-7
703 NBPMCx1E7: CPU Requests to Target Node 0-3/4-7
704 NBPMCx1EB: Request Cache Status 1
704 NBPMCx1F0: Memory Controller Requests
704 NBPMCx3EC: DRAM Accesses
705 NBPMCx3ED: DRAM Controller Page Table Overflows
705 NBPMCx3EE: Memory Controller DRAM Command Slots Missed
705 NBPMCx3EF: Memory Controller Turnarounds
707 IO000: Dma_Ch 0
707 IO002: Dma_Ch 1
707 IO004: Dma_Ch 2
707 IO006: Dma_Ch 3
707 IO008: Dma_Status
707 IO009: Dma_WriteRequest
707 IO00A: Dma_WriteMask
708 IO00B: Dma_WriteMode
708 IO00C: Dma_Clear
708 IO00D: Dma_MasterClr
708 IO00E: Dma_ClrMask
708 IO00F: Dma_AllMask
708 IO020: IntrCntrl1Reg1
708 IO021: IntrCntrl1Reg2
709 IO022: IMCR_Index
709 IO023: IMCR_Data
709 IO040: TimerCh0
709 IO041: TimerCh1
709 IO042: TimerCh2
709 IO043: Tmr1CntrlWord
710 IO060: Keyboard Data
710 IO061: Nmi Status
710 IO064: Keyboard Command Status
711 IO070: RtcAddrPort and NmiMask
711 IO071: RtcDataPort
711 IO072: Alternate RTC AddrPort
711 IO073: Alternate RTC DataPort
711 IO073_x00: RTC Seconds
712 IO073_x01: RTC Seconds Alarm
712 IO073_x02: RTC Minutes
712 IO073_x03: RTC Minutes Alarm
712 IO073_x04: RTC Hours
712 IO073_x05: RTC Hours Alarm
713 IO073_x06: RTC Day of Week
713 IO073_x07: RTC Date of Month
713 IO073_x08: RTC Month
713 IO073_x09: RTC Year
713 IO073_x0A: RTC Register A
714 IO073_x0B: RTC Register B
715 IO073_x0C: RTC Register C
715 IO073_x0D: RTC Date Alarm
715 IO073_x32: RTC AltCentury
715 IO073_x48: RTC Century
716 IO073_x50: RTC Extended RAM Address Port
716 IO073_x53: RTC Extended RAM Data Port
716 IO073_x7E: RTC Time Clear
716 IO073_x7F: RTC RAM Enable
716 IO080: PostCode 0
716 IO081: PostCode 1
717 IO081: Dma_PageCh2
717 IO082: PostCode 2
717 IO082: Dma_PageCh3
717 IO083: PostCode 3
717 IO083: Dma_PageCh1
717 IO084: Dma_Page_Reserved1
717 IO085: Dma_Page_Reserved2
718 IO086: Dma_Page_Reserved3
718 IO087: Dma_PageCh0
718 IO088: Dma_Page_Reserved4
718 IO089: Dma_PageCh6
718 IO08A: Dma_PageCh7
718 IO08B: Dma_PageCh5
718 IO08C: Dma_Page_Reserved5
718 IO08D: Dma_Page_Reserved6
719 IO08E: Dma_Page_Reserved7
719 IO08F: Dma_Refresh
719 IO092: FastInit
719 IO0A0: IntrCntrl2Reg1
719 IO0A1: IntrCntrl2Reg2
719 IO0C0: Dma2_Ch4Addr
719 IO0C2: Dma2_Ch4Cnt
720 IO0C4: Dma2_Ch5Addr
720 IO0C6: Dma2_Ch5Cnt
720 IO0C8: Dma2_Ch6Addr
720 IO0CA: Dma2_Ch6Cnt
720 IO0CC: Dma2_Ch7Addr
720 IO0CE: Dma_Ch7Cnt
720 IO0D0: Dma_Status
720 IO0D2: Dma_WriteRequest
721 IO0D4: Dma_WriteMask
721 IO0D6: Dma_WriteMode
721 IO0D8: Dma_Clear
721 IO0DA: Dma_MasterClr
721 IO0DC: Dma_ClrMask
721 IO0DE: Dma_AllMask
721 IO0F0: NCP Error
722 IO4D0: IntrEdgeControl
723 IOC00: Pci_Intr_Index
723 IOC01: Pci_Intr_Data
723 IOC01_x0[7:0]: PCI INT[H#,G#,F#,E#,D#,C#,B#,A#] Map
724 IOC01_x08: Intr Misc Map
724 IOC01_x09: Intr Misc 0 Map
725 IOC01_x0A: IntrMisc1Map
725 IOC01_x0B: IntrMisc2Map
725 IOC01_x0[7F:C]: PCI Interrupt Map
725 IOC14: Pci_Error
725 IOCD0: PM2_Index
726 IOCD1: PM2_Data
726 IOCD4: BIO SRAMIndex
726 IOCD5: BIO SRAM Data
726 IOCD6: PM_Index
726 IOCD7: PM_Data
726 IOCF9: System Reset Register
728 ABx00: AB Index Register
728 ABx04: AB Data Register
728 ABx04_x54: Misc Control 1
729 ABx04_x58: B-Link RAB Control
729 ABx04_x80: B-Link DMA Prefetch Control
729 ABx04_x90: BIF Control 0
729 ABx04_x94: MSI Control
730 ABx04_x204: SBG Upstream Control
730 ABx04_x10054: A-Link Arbitration Control and Clock Control

730 ABx04_x10090: Misc Control 3
731 D11F0x00: Device/Vendor ID
731 D11F0x04: Status/Command
732 D11F0x08: Revision ID/Class Code
733 D11F0x0C: Header Type Register
733 D11F0x10: Primary IDE CS0 Base Address(BAR0)
734 D11F0x14: Primary IDE CS1 Base Address(BAR1)
734 D11F0x18: Secondary IDE CS0 Base Address (BAR2)
734 D11F0x1C: Secondary IDE CS1 Base Address (BAR3)
734 D11F0x20: Bus Master Interface Register Base Address (BAR4)
735 D11F0x24: AHCI Base Address (BAR5)
735 D11F0x2C: Subsystem ID and Subsystem Vendor ID
735 D11F0x34: Capabilities Pointer
735 D11F0x3C: Interrupt Line
735 D11F0x40: Misc Control
736 D11F0x44: Watch Dog Control And Status
736 D11F0x50: MSI Control
737 D11F0x54: MSI Address
737 D11F0x58: MSI Upper Address
737 D11F0x5C: MSI Data
737 D11F0x60: Power Management Capability
738 D11F0x64: PCI Power Management Control And Status
738 D11F0x70: Serial ATA Capability Register 0
739 D11F0x74: Serial ATA Capability Register 1
739 D11F0x78: IDP Index Register
739 D11F0x7C: IDP Data Register
739 D11F0x7C_x00: HBA Capabilities (CAP)
739 D11F0x7C_x04: Global HBA Control (GHC)
739 D11F0x7C_x08: Interrupt Status (IS)
740 D11F0x7C_x0C: Ports Implemented (PI)
740 D11F0x7C_x10: AHCI Version (VS)
740 D11F0x7C_x14: Command Completion Coalescing Control (CCC_CTL)
740 D11F0x7C_x18: Command Completion Coalescing Ports (CCC_PORTS)
740 D11F0x7C_x1C: Enclosure Management Location (EM_LOC)
740 D11F0x7C_x20: Enclosure Management Control (EM_CTL)
740 D11F0x7C_x24: HBA Capabilities Extended (CAP2)
740 D11F0x7C_x28: BIOS/OS Handoff Control and Status (BOHC)
741 D11F0x7C_x1[8,0]0: Port 1,0 Command List Base Address (PxCLB)
741 D11F0x7C_x1[8,0]4: Port 1,0 Command List Base Upper Address (PxCLBU)
741 D11F0x7C_x1[8,0]8: Port 1,0 FIS Base Address (PxFB)
741 D11F0x7C_x1[8,0]C: Port 1,0 FIS Base Address Upper (PxFBU)
741 D11F0x7C_x1[9,1]0: Port 1,0 Interrupt Status (PxIS)
741 D11F0x7C_x1[9,1]4: Port 1,0 Interrupt Enable (PxIE)
741 D11F0x7C_x1[9,1]8: Port 1,0 Command and Status (PxCMD)
741 D11F0x7C_x1[A,2]0: Port 1,0 Task File Data (PxTFD)
742 D11F0x7C_x1[A,2]4: Port 1,0 Signature (PxSIG)
742 D11F0x7C_x1[A,2]8: Port 1,0 Serial ATA Status (PxSSTS)
742 D11F0x7C_x1[A,2]C: Port 1,0 Serial ATA Control (PxSCTL)
742 D11F0x7C_x1[B,3]0: Port 1,0 Serial ATA Error (PxSERR)
742 D11F0x7C_x1[B,3]4: Port 1,0 Serial ATA Active (PxSACT)
742 D11F0x7C_x1[B,3]8: Port 1,0 Command Issue (PxCI)
742 D11F0x7C_x1[B,3]C: Port 1,0 SNotification (PxSNTF)
742 D11F0x7C_x1[C,4]0: Port 1,0 FIS-based Switching Control (PxFBS)
743 D11F0x80: PHY Core Control 1
743 D11F0x84: PHY Core Control 2
744 D11F0x88: PHY Global Control 1
744 D11F0x8C: PHY Global Control 2
744 D11F0x98: PHY Fine Tune PortX GenX Setting
744 D11F0x9C: PortX Setting 2
745 D11F0xB4: PortX BIST Control/Status
746 D11F0xB7: PortX BIST Port Select
746 D11F0xBC: T-Mode BIST Transit Pattern DW1
746 D11F0xC0: T-Mode BIST Transit Pattern DW2
746 D11F0xC4: T-Mode BIST Transit Control
746 D11F0xD0: Advanced Features Capability Register 0
747 D11F0xD4: Advanced Features Capability Register 1
747 D11F0xE0: PCI Target Control TimeOut Counter
748 IDE[1:0]x00: IDE DATA
748 IDE[1:0]x01: Feature and Error
748 IDE[1:0]x02: Sector Count
748 IDE[1:0]x03: Sector Number
748 IDE[1:0]x04: Cylinder Low
749 IDE[1:0]x05: Cylinder High
749 IDE[1:0]x06: Drive and Head
749 IDE[1:0]x07: Command and Status
749 IDEA[1:0]x02: Device Control and Alternate Status
749 IDE_BMx0[8,0]: Bus Master IDE Command
750 IDE_BMx0[A,2]: Bus-master IDE Status
750 IDE_BMx0[C,4]: Descriptor Table Pointer
751 SATAx00: HBA Capabilities (CAP)
752 SATAx04: Global HBA Control (GHC)
753 SATAx08: Interrupt Status (IS)
754 SATAx0C: Ports Implemented (PI)
754 SATAx10: AHCI Version (VS)
754 SATAx14: Command Completion Coalescing Control (CCC_CTL)
755 SATAx18: Command Completion Coalescing Ports (CCC_PORTS)
755 SATAx1C: Enclosure Management Location (EM_LOC)
755 SATAx20: Enclosure Management Control (EM_CTL)
756 SATAx24: HBA Capabilities Extended (CAP2)
756 SATAx28: BIOS/OS Handoff Control and Status (BOHC)
757 SATAx1[8,0]0: Port 1,0 Command List Base Address (PxCLB)
757 SATAx1[8,0]4: Port 1,0 Command List Base Upper Address (PxCLBU)
757 SATAx1[8,0]8: Port 1,0 FIS Base Address (PxFB)
758 SATAx1[8,0]C: Port 1,0 FIS Base Address Upper (PxFBU)
758 SATAx1[9,1]0: Port 1,0 Interrupt Status (PxIS)
759 SATAx1[9,1]4: Port 1,0 Interrupt Enable (PxIE)
761 SATAx1[9,1]8: Port 1,0 Command and Status (PxCMD)
763 SATAx1[A,2]0: Port 1,0 Task File Data (PxTFD)
764 SATAx1[A,2]4: Port 1,0 Signature (PxSIG)
764 SATAx1[A,2]8: Port 1,0 Serial ATA Status (PxSSTS)
765 SATAx1[A,2]C: Port 1,0 Serial ATA Control (PxSCTL)
766 SATAx1[B,3]0: Port 1,0 Serial ATA Error(PxSERR)
768 SATAx1[B,3]4: Port 1,0 Serial ATA Active (PxSACT)
769 SATAx1[B,3]8: Port 1,0 Command Issue (PxCI)
769 SATAx1[B,3]C: Port 1,0 SNotification (PxSNTF)
769 SATAx1[C,4]0: Port 1,0 FIS-based Switching Control (PxFBS)
770 SATAx1[C,4]4: Port 1,0 Device Sleep (PxDEVSLP)
771 SATA_EMx00: Message Header
772 SATA_EMx04: Write SGPIO Register Request (I)
772 SATA_EMx08: Write SGPIO Register Request (II)
773 SATA_EMx0C: Write SGPIO Register Request (III)
774 D12F0x00: Device/Vendor ID
774 D12F0x04: Status/Command
775 D12F0x08: Revision ID / Class Code
775 D12F0x0C: Miscellaneous
776 D12F0x10: BAR_EHCI
776 D12F0x2C: Subsystem ID / Subsystem Vendor ID
776 D12F0x34: Capability Pointer
776 D12F0x3C: Interrupt Line
776 D12F0x50: EHCI Misc Control
778 D12F0x54: EHCI Spare 1
779 D12F0x60: Serial Bus Release Number / FLADJ
779 D12F0x64: Misc Control 2

780 D12F0x70: Over-Current Control 1
780 D12F0x74: Misc Control 2
780 D12F0x78: Target Timeout Control
781 D12F0x80: Hub Config 0
781 D12F0x84: Hub Config 1
782 D12F0x88: HUB Config 2
782 D12F0x8C: HUB Config 3
783 D12F0x9C: HUB Status
783 D12F0xA0: USB Legacy Support Extended Capability
784 D12F0xA4: USB Legacy Support Control / Status
785 D12F0xC0: PME Capability
785 D12F0xC4: PME Control / Status
786 D12F0xD0: MSI Control
786 D12F0xD4: MSI Address
786 D12F0xD8: MSI Upper Address
786 D12F0xDC: MSI Data
787 D12F0xF0: Function Level Reset Capability
787 D12F0xF4: Function Level Reset Control
788 EHCI1x00: Capability Length
788 EHCI1x02: HC Interface Version
788 EHCI1x04: HC Structural Parameters
789 EHCI1x08: HC Capability Parameters
790 EHCI1x20: USB Command
791 EHCI1x24: USB Status
793 EHCI1x28: USB Interrupt Enable
793 EHCI1x2C: Frame Index
794 EHCI1x30: Control Data Structure Segment
794 EHCI1x34: Periodic Frame List Base Address
794 EHCI1x38: Current Async List Address
794 EHCI1x60: Configure Flag
794 EHCI1x[70,6C,68,64]: Port Status Control [4:1]
797 EHCI1xA4: Packet Buffer Threshold Values
797 EHCI1xB4: UTMI Control
798 EHCI1xB8: Loopback Test
799 EHCI1xBC: EOR MISC Control
799 EHCI1xC0: USB Common PHY CAL and Control
800 EHCI1xC4: USB Common PHY Control 1
800 EHCI1xD0: USB Common PHY Control 2
801 EHCI1xD4: USB Common PHY Control 3
801 EHCI1xDC: USB Battery Charger Enable
801 D10F0x00: Device/Vendor ID
801 D10F0x04: Status/Command
802 D10F0x08: Revision ID / Class Code
803 D10F0x0C: Miscellaneous
803 D10F0x10: Bar 0
803 D10F0x14: Bar 1
803 D10F0x2C: Subsystem Vendor ID / Subsystem ID
803 D10F0x34: Capability Pointer
804 D10F0x3C: Interrupt Line
804 D10F0x40: IDP Index Register
804 D10F0x44: IDP Data Register
804 D10F0x48: Indirect PCI Index Register
805 D10F0x4C: Indirect PCI Data Register
805 D10F0x4C_x08: Port Disable Write Once
805 D10F0x4C_x0C: Port Disable RW
805 D10F0x4C_x10: USB DCLK Event Counter 0
806 D10F0x4C_x14: USB DCLK Event Counter 1
806 D10F0x4C_x18: USB DCLK Event Counter Select
806 D10F0x4C_x1C: USB DCLK Event Counter Control
807 D10F0x4C_x4000_0000: UTMI Control
807 D10F0x4C_x4000_0004: USB PHY Status
807 D10F0x4C_x4000_0018: USB Common PHY Calibration and Control
808 D10F0x4C_x4000_001C: USB Common PHY Control
808 D10F0x4C_x4000_0020: HS Loopback Test
809 D10F0x4C_x4000_0024: CL Loopback Control
809 D10F0x4C_x4000_0028: Misc Control
810 D10F0x4C_x4000_002C: AMDU2IF_POWERUP_CHECK
810 D10F0x4C_x4000_0030: USB SCLK Event Counter 0
810 D10F0x4C_x4000_0034: USB SCLK Event Counter 1
810 D10F0x4C_x4000_0038: USB SCLK Event Counter Select
810 D10F0x4C_x4000_003C: USB SCLK Event Counter Control
811 D10F0x4C_x4000_0058: LPM Control
811 D10F0x4C_x4000_0060: USB Common PHY Control 2
811 D10F0x4C_x4000_0064: USB Common PHY Control 3
812 D10F0x50: PME Capability
812 D10F0x54: PME Control / Status
813 D10F0x60: SBRN
814 D10F0x70: MSI Control
814 D10F0x74: MSI Address
814 D10F0x78: MSI Upper Address
814 D10F0x7C: MSI Data
815 D10F0x80: MSI Mask Bits
815 D10F0x90: MSI-X Control
815 D10F0x94: MSI-X Table Offset/Table BIR
815 D10F0x98: MSI-X PBA Offset/PBA BIR
816 D10F0xA0: PCIe® Capability List
816 D10F0xA4: Device Capability
817 D10F0xA8: Device Control/Status
818 D10F0xC4: Device Capabilities 2
819 D10F0xC8: Device Control/Status 2
819 D10F0x100: LTR Extended Capability Header
819 D10F0x104: Max Latency
820 XHCI_CAPx02: HC Interface Version
820 XHCI_PMx00: xHCI Config 0
821 XHCI_PMx04: xHCI Firmware Addr 0
821 XHCI_PMx08: xHCI Firmware Addr 1
821 XHCI_PMx10: xHCI Memory Config
822 XHCI_PMx14: PLL Control
822 XHCI_PMx18: USB2.0 Link State
823 XHCI_PMx20: USB2.0 Wake Control
823 XHCI_PMx2C: xHCI Misc 2
825 XHCI_PMx30: xHCI 1.0 Enable
825 XHCI_PMx48: SSPHY ACPI Indirect Index
826 XHCI_PMx4C: SSPHY ACPI IndirectData
826 XHCI_PMx4C_x30: Over current Control
826 XHCI_PMx4C_x2[C,8,4,0]0: SSPHY Port[3:0] Test Control 0
826 XHCI_PMx8C: SSPHY Common Control 0
827 XHCI_PMxA0: SPI BAR0
827 XHCI_PMxA4: SPI BAR1
827 XHCI_PMxA8: SPI BAR2
827 XHCI_PMxAC: SPI BAR3
827 XHCI_PMxB0: SPI Valid Base
828 XHCI_PMxB4: SPI Misc
828 XHCI_PMxB8: FW_DMA_ADDR_LOW
828 XHCI_PMxBC: FW_DMA_ADDR_HIGH
828 XHCI_PMx[FC:C0]: SPI Data Block N
829 HCEx40: HCE Control
829 HCEx44: HCE Input
830 HCEx48: HCE Output
830 HCEx4C: HCE Status
831 HCEx50: HCE IntrEn
832 D14F7x00: Device/Vendor ID
832 D14F7x04: Status/Command
833 D14F7x08: Revision ID/Class Code
833 D14F7x0C: Cache Line Size
833 D14F7x10: Base Address Reg 0

833 D14F7x14: Upper Base Address Reg 0
834 D14F7x2C: Subsystem ID and Subsystem Vendor ID
834 D14F7x34: Capabilities Pointer
834 D14F7x3C: Interrupt Line
834 D14F7x40: Slot Information
835 D14F7x80: SD PCI MSI Capability Header
835 D14F7x84: SD PCI MSI Address
835 D14F7x88: SD PCI MSI Upper Address
835 D14F7x8C: SD PCI MSI Data
836 D14F7x90: Power Management Capability Header
836 D14F7x94: Power Management Control and Status Register
837 D14F7xB8: SD Auto Pattern
837 SDHC0x00: SDHC System Address / Argument 2
838 SDHC0x04: SDHC Block CS
838 SDHC0x08: SDHC Command Argument
838 SDHC0x0C: SDHC Command/Transfer Mode
839 SDHC0x10: SDHC_RESP1_0
839 SDHC0x14: SDHC_RESP3_2
839 SDHC0x18: SDHC_RESP5_4
840 SDHC0x1C: SDHC_RESP7_6
840 SDHC0x20: SDHC_BUFFER
840 SDHC0x24: SDHC_PRSNT_STATE
841 SDHC0x28: SDHC_CTRL1
842 SDHC0x2C: SDHC_CTRL2
843 SDHC0x30: SDHC_INT_STATUS
844 SDHC0x34: SDHC_INT_MASK
845 SDHC0x38: SDHC_SIG_MASK
846 SDHC0x3C: SDHC ACMD12 Error/Host Control 2
846 SDHC0x40: SDHC_CAPABILITY
847 SDHC0x44: SDHC_CAPABILITY 2
848 SDHC0x48: SDHC_CURR_CAPABILITY
848 SDHC0x50: SDHC_FORCE_EVT
849 SDHC0x54: SDHC_ADMA_ERR
849 SDHC0x58: SDHC_ADMA_SAD
849 SDHC0x6[C:0:step4]: Preset Value
850 SDHC0xE0: Shared Bus Control Register
851 SDHC0xFC: SDHC_VER_SLOT
853 D14F0x00: Device/Vendor ID
853 D14F0x04: Status/Command
854 D14F0x08: Revision ID/Class Code
854 D14F0x0C: Cache Line Size
854 D14F0x10: Base Address 0
854 D14F0x14: Base Address 1
855 D14F0x18: Base Address 2
855 D14F0x1C: Base Address 3
855 D14F0x20: Base Address 4
855 D14F0x24: Base Address 5
855 D14F0x28: Cardbus CIS Pointer
855 D14F0x2C: Subsystem Vendor ID
855 D14F0x30: Expansion ROM Base Address
856 D14F0x34: Capability Pointer
856 D14F0x3C: Interrupt Line
856 D14F0xFC: ScratchCfgReg
856 ASFx00: HostStatus
857 ASFx02: HostControl
857 ASFx03: HostCommand
857 ASFx04: SlaveAddress
857 ASFx05: Data0
858 ASFx06: Data1
858 ASFx07: DataIndex
858 ASFx08: PEC
858 ASFx09: ListenAdr
858 ASFx0A: ASFStatus
859 ASFx0B: StatusMask0
859 ASFx0C: StatusMask1
859 ASFx0D: SlaveStatus
859 ASFx0E: RemoteCtrlAdr
859 ASFx0F: SensorAdr
859 ASFx10: DataReadPointer
860 ASFx11: DataWritePointer
860 ASFx12: SetDataReadPointer
860 ASFx13: DataBankSel
860 ASFx14: Semaphore
861 ASFx15: SlaveEn
861 ASFx16: DelayMasterTimer
861 SMBUSx00: SMBusStatus
862 SMBUSx01: SMBusSlaveStatus
862 SMBUSx02: SMBusControl
863 SMBUSx03: SMBusHostCmd
863 SMBUSx04: SMBusAddress
863 SMBUSx05: SMBusData0
863 SMBUSx06: SMBusData1
863 SMBUSx07: SMBusBlockData
864 SMBUSx08: SMBusSlaveControl
864 SMBUSx09: SMBusShadowCmd
864 SMBUSx0A: SMBusSlaveEvent
865 SMBUSx0C: SlaveData
865 SMBUSx0E: SMBusTiming
865 SMBUSx10: I2CbusConfig
865 SMBUSx11: I2CCommand
865 SMBUSx12: I2CShadow1
865 SMBUSx13: I2Cshadow2
866 SMBUSx14: SMBusAutoPoll
866 SMBUSx15: SMBusCounter
866 SMBUSx16: SMBusStop
867 SMBUSx17: SMBusHostCmd2
868 IOAPICx00: IO Register Select Register
868 IOAPICx10: IO Window Register
868 IOAPICx10_x00: IOAPIC ID Register
868 IOAPICx10_x01: IOAPIC Version Register
868 IOAPICx10_x02: IOAPIC Arbitration Register
869 IOAPICx10_x[3E:10:step2]: Redirection Table Entry [23:0]
869 IOAPICx20: IRQ Pin Assertion Register
869 IOAPICx40: EOI Register
870 D14F3x00: Device/Vendor ID
870 D14F3x04: Status/Command
871 D14F3x08: Revision ID/Class Code
871 D14F3x0C: Cache Line Size
871 D14F3x10: Base Address Reg 0
871 D14F3x2C: Subsystem ID and Subsystem Vendor ID
871 D14F3x34: Capabilities Pointer
872 D14F3x40: PCI Control
872 D14F3x44: IO Port Decode Enable
874 D14F3x48: IO or Memory Port Decode Enable
875 D14F3x4C: Memory Range
875 D14F3x[5C,58,54,50]: ROM Protect 3, 2, 1, 0
875 D14F3x60: PCI Memory Address for LPC Target Cycles
876 D14F3x64: PCI IO base Address for Wide Generic Port
876 D14F3x68: ROM Address Range 1
876 D14F3x6C: ROM Address Range 2
876 D14F3x74: Alternative Wide IO Range Enable
877 D14F3x78: Miscellaneous Control Bits
877 D14F3x7C: Trusted Platform Module (TPM)
878 D14F3x84: TMKBC_BaseAddrLow
878 D14F3x88: TMKBC_BaseAddrHigh
879 D14F3x8C: TMKBC_Remap

879 D14F3x90: Wide IO 2
879 D14F3x98: EC_LPC_Cntrl
879 D14F3xA0: SPI Base_Addr
880 D14F3xA4: EC_PortAddress
880 D14F3xB0: RomDmaSrcAddr
880 D14F3xB4: RomDmaDstAddr
880 D14F3xB8: RomDmaControl/EcControl/HostControl
881 D14F3xC0: EcRomWrOffset
881 D14F3xC4: EcRomRdOffset
882 D14F3xC8: ClientRomProtect
882 D14F3xCC: AutoRomCfg
882 D14F3xD0: ClkCntrl
883 D14F3xD4: ClkRunOption
883 SPIx00: SPI_Cntrl0
884 SPIx04: SPI_RestrictedCmd
885 SPIx08: SPI_RestrictedCmd2
885 SPIx0C: SPI_Cntrl1
885 SPIx10: SPI_CmdValue0
886 SPIx14: SPI_CmdValue1
886 SPIx18: SPI_CmdValue2
887 SPIx1C: Reserved
887 SPIx1D: Alt_SPI_CS
887 SPIx1E: SpiExtRegIndx
887 SPIx1F: SpiExtRegData
888 SPIx1F_x00: DDR_CMD
888 SPIx1F_x01: QDR_CMD
888 SPIx1F_x02: DPR_CMD
888 SPIx1F_x03: QPR_CMD
888 SPIx1F_x04: ModeByte
888 SPIx1F_x05: TxByteCount
888 SPIx1F_x06: RxByteCount
889 SPIx1F_x07: SPIDataFifoPtr
889 SPIx20: SPI100 Enable
889 SPIx22: SPI100 Speed Config
890 SPIx2C: SPI100 Host Prefetch Config
891 SPIx40: DDRCmdCode
891 SPIx41: QDRCmdCode
891 SPIx42: DPRCmdCode
891 SPIx43: QPRCmdCode
891 SPIx44: ModeByte
891 SPIx45: CmdCode
892 SPIx47: CmdTrigger
892 SPIx48: TxByteCount
892 SPIx4B: RxByteCount
892 SPIx4C: SpiStatus
892 SPIx[C6:80]: FIFO[70:0]
893 eSPIx00: eSPI Software Specific Register 0
894 eSPIx04: eSPI Software Specific Register 1
895 eSPIx[1C:08:Step6]: eSPI Software Specific Register 2-7
896 eSPIx20: eSPI Software Specific Register 8
897 eSPIx24: eSPI Software Specific Register 9
898 eSPIx28: eSPI Software Specific Register 10
898 eSPIx2C: eSPI Master Capability
899 eSPIx30: eSPI Global Control and Status Register 0
900 eSPIx34: eSPI Global Control and Status Register 1
902 eSPIx38: eSPI MISC Control Register 0
902 eSPIx3C: eSPI MISC Control Register 1
902 eSPIx40: eSPI IO or MMIO Decoding Enable Register for Slave N
903 eSPIx44: eSPI IO Target Range Register 0 for Slave N
903 eSPIx48: eSPI IO Target Range Register 1 for Slave N
903 eSPIx4C: eSPI IO Target Range Register 2 for Slave N
903 eSPIx50: eSPI MMIOTarget Range Register 0 for Slave N
903 eSPIx54: eSPI MMIOTarget Range Register 1 for Slave N
903 eSPIx58: eSPI MMIOTarget Range Register 2 for Slave N
903 eSPIx5C: eSPI MMIOTarget Range Register 3 for Slave N
904 eSPIx60: eSPI MMIOTarget Range Register 4 for Slave N
904 eSPIx64: eSPI MMIOTarget Range Register 5 for Slave N
904 eSPIx68: eSPI Slave N Configuration
905 eSPIx6C: eSPI Slave N Interrupt Enable
905 eSPIx70: eSPI Slave N Interrupt Status
906 eSPIx74: eSPI Slave N Received Peripheral Message Register 0
907 eSPIx78: eSPI Slave N Received Peripheral Message Register 1
907 eSPIx[98:7C:Step8]: eSPI Slave N Received Peripheral Message Register 2-9
907 eSPIx9C: eSPI Slave N Received Virtual Wires Register
908 eSPIxA0: eSPI Slave N Virtual Wire Received Data Register
908 eSPIxA4: eSPI Slave N Virtual Wire Index Selection Register
908 eSPIxA8: eSPI Slave N Virtual Wire MISC Control Register
910 eSPIxAC: eSPI Slave N Virtual Wires Polarity Register
910 eSPIxB0: eSPI Slave N Received OOB Message Register 0
910 eSPIxB4: eSPI Slave N Received OOB Message Register 1
910 eSPIx[D0:B8:Step7]: eSPI Slave N Received OOB Message Register 2-8
911 eSPIxD4: eSPI Slave N Received OOB Message Register 9
911 eSPIxD8: eSPI Slave N Reserved Register 0
911 eSPIxDC: eSPI Slave N Reserved Register 1
912 HPETx000: ID
912 HPETx004: ClkPeriod
912 HPETx010: Config
912 HPETx020: Interrupt Status
913 HPETx0F0: Main Counter
913 HPETx1[4:0:Step2]0: Timer[2:0] Config Capability
914 HPETx1[4:0:Step2]8: Timer[2:0] Comparator
914 HPETx1[5:1:step2]0: Timer[2:0] FSB Interrupt Data
914 HPETx1[5:1:step2]4: Timer[2:0] FSB Interrupt Address
914 HPETx1[D:B]0: Timer[2:0] Comparator Base Shadow
915 HPETx1[D:B]8: Timer[2:0] Comparator Shadow
915 HPETx1E0: Main Counter RTC
915 HPETx1E8: Next Timer Remain
916 MISCx00: GPPClkCntrl
918 MISCx04: ClkOutputCntrl
920 MISCx08: CGPLLConfig1
921 MISCx0C: CGPLLConfig2
922 MISCx10: CGPLLConfig3
923 MISCx14: CGPLLConfig4
923 MISCx18: CGPLLConfig5
923 MISCx1C: CGPLLConfig6
924 MISCx20: IMP Calibration
925 MISCx24: ClkDrvStr1
926 MISCx28: ClkDrvStr2
927 MISCx2C: ClkGatedCntl
928 MISCx30: CGPLLConfig7
928 MISCx34: CGPLLConfig8
928 MISCx38: CGPLLConfig9
928 MISCx3C: CGPLLConfig10
928 MISCx40: MiscClkCntl1
930 MISCx44: MiscClkCntl2
930 MISCx48: MiscClkCntl3
931 MISCx4C: MiscClkCntl4
931 MISCx60: IdleCntrl
931 MISCx68: Memory Power Saving Control
931 MISCx6C: MiscControl
932 MISCx70: OscFreqCounter
932 MISCx74: HpetClkPeriod
932 MISCx78: PostCode
932 MISCx80: StrapStatus
933 MISCx90: AutoTransaction/Allow EC

933 MISCx94: AutoAddrLow
933 MISCx98: AutoAddrHigh
934 MISCx9C: AutoData
934 MISCxC0: CPU Pstate0
934 MISCxC4: CPU Pstate1
934 MISCxD0: CPU Cstate0
935 MISCxD4: CPU Cstate1
935 MISCxF0: SataPortSts
935 MISCxF4: ClkCntrlSts
936 GPIOx[0F8:000:step4]: GPIO Bank 0 Control Register
938 GPIOxFC: GPIO_Wake_Inter_Master Switch
939 GPIOx[1FC:100:step4]: GPIO Bank 1 Control Register
941 GPIOx[2DC:200:step4]: GPIO Bank 2 Control Register
943 GPIOx2F0: GPIO_Wake_Status_Index_0
943 GPIOx2F4: GPIO_Wake_Status_Index_1
943 GPIOx2F8: GPIO_Interrupt_Status_Index_0
943 GPIOx2FC: GPIO_Interrupt_Status_Index_1
944 IOMUXx00: PWR_BTN_L_AGPIO0
944 IOMUXx01: SYS_RESET_L_AGPIO1
944 IOMUXx02: WAKE_L_AGPIO2
945 IOMUXx03: AGPIO3
945 IOMUXx04: AGPIO4
945 IOMUXx05: AGPIO5_DEVSLP0_S5
946 IOMUXx06: AGPIO6_LDT_RST_L
946 IOMUXx07: AGPIO7_LDT_PWROK
946 IOMUXx08: AGPIO8_SerPortTX_OUT
947 IOMUXx09: AGPIO9_SerPortRX_OUT
947 IOMUXx0A: S0I3_GPIO_AGPIO10
947 IOMUXx0B: BLINK _AGPIO11_USB_OC7_L_AGPIO11
948 IOMUXx0C: IR_LED_L_LLB_L_AGPIO12
948 IOMUXx0D: IR_TX0_USB_OC5_L_AGPIO13
948 IOMUXx0E: IR_TX1_USB_OC6_L_AGPIO14
949 IOMUXx0F: IR_RX1_AGPIO15
949 IOMUXx10: USB_OC0_L_TRST_L_AGPIO16
949 IOMUXx11: USB_OC1_L_TDI_AGPIO17
950 IOMUXx12: USB_OC2_L_TCK_AGPIO18
950 IOMUXx13: SCL1_I2C3_SCL_AGPIO19
950 IOMUXx14: SDA1_I2C3_SDA_AGPIO20
951 IOMUXx15: LPC_PD_L_AGPIO21
951 IOMUXx16: LPC_PME_L_AGPIO22
951 IOMUXx17: AC_PRES_USB_OC4_L_IR_RX0_AGPIO23
952 IOMUXx18: TDO_USB_OC3_L_AGPIO24
952 IOMUXx19: SD0_CD_AGPIO25
952 IOMUXx1A: PCIE_RST_L_EGPIO26
953 IOMUXx27: VDDGFX_PD_AGPIO39
953 IOMUXx28: AGPIO40
953 IOMUXx2A: S5_MUX_CTRL_EGPIO42
954 IOMUXx40: AGPIO64
954 IOMUXx41: AGPIO65
954 IOMUXx42: AGPIO66_ShutDown_L
955 IOMUXx43: EGPIO67_DEVSLP0
955 IOMUXx44: AGPIO68_SGPIO_CLK
955 IOMUXx45: AGPIO69_SGPIO_LOAD
956 IOMUXx46: EGPIO70_DEVSLP1
956 IOMUXx47: AGPIO71_SGPIO_DATAOUT
956 IOMUXx48: AGPIO72_SGPIO_DATAIN
957 IOMUXx4A: LPCCLK0_EGPIO74
957 IOMUXx4B: LPCCLK1_EGPIO75
957 IOMUXx4C: AGPIO76_SPI_TPM_CS_L
958 IOMUXx54: FANIN0_AGPIO84
958 IOMUXx55: FANOUT0_AGPIO85
958 IOMUXx56: LPC_SMI_L_AGPIO86
959 IOMUXx57: SERIRQ_AGPIO87
959 IOMUXx58: LPC_CLKRUN_L_AGPIO88
959 IOMUXx59: GENINT1_L_AGPIO89
960 IOMUXx5A: GENINT2_L_AGPIO90
960 IOMUXx5B: SPKR_AGPIO91
960 IOMUXx5C: CLK_REQ0_L_SATA_IS0_L_SATA_ZP0_L_AGPIO92
961 IOMUXx5D: SD0_LED_EGPIO93
961 IOMUXx5F: SD0_CLK_EGPIO95
961 IOMUXx60: SD0_CMD_EGPIO96
962 IOMUXx[64:61]: SD0_DATA[3:0]_EGPIO[100:97]
962 IOMUXx65: SD0_WP_EGPIO101
962 IOMUXx66: SD0_PWR_CTRL_AGPIO102
963 IOMUXx71: SCL0_I2C2_SCL_EGPIO113
963 IOMUXx72: SDA0_I2C2_SDA_EGPIO114
963 IOMUXx73: CLK_REQ1_L_AGPIO115
964 IOMUXx74: CLK_REQ2_L_AGPIO116
964 IOMUXx75: XHC_SPI_CLK_SPI_CLK_ESPI_CLK_EGPIO117
964 IOMUXx76: XHC_SPI_CS1_L_SPI_CS1_L_EGPIO118
965 IOMUXx77: SPI_CS2_L_ESPI_CS_L_EGPIO119
965 IOMUXx78: XHC_SPI_DI_SPI_DI_ESPI_DAT1_EGPIO120
965 IOMUXx79: XHC_SPI_DO_SPI_DO_ESPI_DAT0_EGPIO121
966 IOMUXx7A: SPI_WP_L_ESPI_DAT2_EGPIO122
966 IOMUXx7E: GA20IN_AGPIO126
966 IOMUXx81: KBRST_L_AGPIO129
967 IOMUXx82: SATA_ACT_L_AGPIO130
967 IOMUXx83: CLK_REQ3_L_SATA_IS1_L_SATA_ZP1_L_EGPIO131
967 IOMUXx84: CLK_REQG_L_OSCIN_EGPIO132
968 IOMUXx85: SPI_HOLD_L_ESPI_DAT3_EGPIO133
968 IOMUXx87: UART0_CTS_L_EGPIO135
968 IOMUXx88: UART0_RXD_EGPIO136
969 IOMUXx89: UART0_RTS_L_EGPIO137
969 IOMUXx8A: UART0_TXD_EGPIO138
969 IOMUXx8B: UART0_INTR_AGPIO139
970 IOMUXx8C: UART1_CTS_L_EGPIO140
970 IOMUXx8D: UART1_RXD_EGPIO141
970 IOMUXx8E: UART1_RTS_L_EGPIO142
971 IOMUXx8F: UART1_TXD_EGPIO143
971 IOMUXx90: UART1_INTR_AGPIO144
971 IOMUXx91: I2C0_SCL_EGPIO145
972 IOMUXx92: I2C0_SDA_EGPIO146
972 IOMUXx93: I2C1_SCL_EGPIO147
972 IOMUXx94: I2C1_SDA_EGPIO148
973 PMx00: DecodeEn
974 PMx04: IsaControl
975 PMx08: PciControl
976 PMx0C: StpClkSmaf
977 PMx10: Power Reset Config
978 PMx40: eSPIIntrCtrl
978 PMx44: BootTimerEn
978 PMx48: PGPwrEnDly
978 PMx54: SerialIrqConfig
979 PMx56: RTC Control
980 PMx58: VRT_T1
980 PMx59: VRT_T2
980 PMx5B: RTC Shadow
980 PMx5C: LLBCntrl
980 PMx5E: RTC ExtIndex
981 PMx5F: RTC ExtData
981 PMx5F_x00: RTCEXT DltSavEnable
981 PMx5F_x01: RTCEXT SprFwdCtrl
981 PMx5F_x02: RTCEXT SprFwdMonth
981 PMx5F_x03: RTCEXT FallBackCtrl
982 PMx5F_x04: RTCEXT FallBackMonth
982 PMx5F_x10: RTCEXT WeekTimerControl

982 PMx5F_x11: RTCEXT WeekTimerReloadLow
982 PMx5F_x12: RTCEXT WeekTimerReloadHigh
983 PMx5F_x13: RTCEXT WeekTimerDataLow
983 PMx5F_x14: RTCEXT WeekTimerDataHigh
983 PMx60: AcpiPm1EvtBlk
983 PMx62: AcpiPm1CntBlk
983 PMx64: AcpiPmTmrBlk
983 PMx66: CpuCntBlk
984 PMx68: AcpiGpe0Blk
984 PMx6A: AcpiSmiCmd
984 PMx6E: AcpiPm2CntBlk
984 PMx74: AcpiConfig
985 PMx78: WakeIoAddr
985 PMx7A: HaltCountEn
985 PMx7C: C1eWrPortAdr
985 PMx7E: CStateEn
986 PMx80: Break Event
987 PMx88: CStateControl
987 PMx8E: PopUpEndTime
987 PMx94: CStateTiming0
988 PMx98: CStateTiming1
988 PMx9C: C2Count
988 PMx9D: C3Count
989 PMxA0: MessageCState
990 PMxB0: DeferTimeTick
990 PMxB4: Tpreset1b
991 PMxB8: Tpreset2
991 PMxB9: LpcMisc
991 PMxBA: S_StateControl
992 PMxBC: ThrottlingControl
992 PMxBE: ResetControl1
993 PMxC0: S5/Reset Status
994 PMxC4: ResetCommand
994 PMxC5: CF9 Shadow
995 PMxC8: Misc
996 PMxCC: IoDrvSth
996 PMxD0: RstCntrl
996 PMxD1: Reset Function
996 PMxD1_x0: RstLength
996 PMxD1_x1: APURstLength
997 PMxD1_x3: APUPwrGdLength
997 PMxD2: Pmio
997 PMxD6: IMC Gating
997 PMxD8: Eprom Index
997 PMxD9: Eprom Data
997 PMxDA: SataConfig
998 PMxDC: SataConfig2
998 PMxE0: ABRegBar
999 PMxE4: AB Misc Control
999 PMxE8: SDFlashCntrl
999 PMxEB: AzEn
999 PMxEC: LpcGating
999 PMxED: USB Gating
1000 PMxEF: USB Enable
1000 PMxF0: USB Control
1001 PM2x00: Fan0InputControl
1002 PM2x01: Fan0Control
1002 PM2x02: Fan0Freq
1003 PM2x03: LowDuty0
1003 PM2x04: MedDuty0
1003 PM2x05: Multiplier0
1004 PM2x06: LowTemp0Lo
1004 PM2x07: LowTemp0Hi
1004 PM2x08: MedTemp0Lo
1004 PM2x09: MedTemp0Hi
1004 PM2x0A: HighTemp0Lo
1004 PM2x0B: HighTemp0Hi
1005 PM2x0C: LinearRange0
1005 PM2x0D: LinearHoldCount0
1005 PM2x0E: Fan0Hysteresis
1005 PM2x50: Med2Temp0Lo
1005 PM2x51: Med2Temp0Hi
1005 PM2x52: Med2Duty0
1005 PM2x53: Multiplier2_0
1006 PM2x60: FanStatus
1006 PM2x61: FanINTRouteLo
1006 PM2x63: SampleFreqDiv
1007 PM2x64: FanDebounceCounterLo
1007 PM2x65: FanDebounceCounterHi
1007 PM2x66: Fan0DetectorControl
1007 PM2x67: Fan0SpeedLimitLo
1007 PM2x68: Fan0SpeedLimitHi
1008 PM2x69: Fan0SpeedLo
1008 PM2x6A: Fan0SpeedHi
1008 PM2x8A: TempTsiLo
1008 PM2x8B: TempTsiHi
1008 PM2x8C: TempTsiLimitLo
1008 PM2x8D: TempTsiLimitHi
1008 PM2x8E: TempTsiChangeLimit
1009 PM2x8F: TempTsiWe
1009 PM2x90: TempTsiStatus
1009 PM2x92: TempTsiControl
1009 PM2x94: TempTsiINTRoute
1010 PM2xDF: TempTsiRstSel
1010 PM2xE0: AlertThermaltripStatus
1010 PM2xE1: AlertLimitLo
1010 PM2xE2: AlertLimitHi
1010 PM2xE3: ThermalTripLimitLo
1011 PM2xE4: ThermalTripLimitHi
1011 PM2xE5: AlertThermaltripControl
1011 AcpiPmEvtBlkx00: Pm1Status
1012 AcpiPmEvtBlkx02: Pm1Enable
1012 AcpiPm1CntBlkx00: PmControl
1013 AcpiPm2CntBlkx00: Pm2Control
1013 AcpiPmTmrBlkx00: TmrValue/ETmrValue
1013 CpuCntBlkx00: ClkValue
1014 CpuCntBlkx04: PLvl2
1014 CpuCntBlkx05: PLvl3
1014 AcpiGpe0Blkx00: EventStatus
1014 AcpiGpe0Blkx04: EventEnable
1014 SmiCmdBlkx00: SmiCmdPort
1014 SmiCmdBlkx01: SmiCmdStatus
1015 SMIx00: Event_Status
1015 SMIx04: Event_Enable
1015 SMIx08: SciTrig
1017 SMIx0C: SciLevl
1018 SMIx10: SmiSciStatus
1018 SMIx14: SmiSciEn
1018 SMIx18: SwSciEn
1018 SMIx1C: SwSciData
1018 SMIx20: SciSleepDisable
1019 SMIx30: CapturedData
1019 SMIx34: CapturedValid
1019 SMIx38: EPBIF_AER_Straps
1020 SMIx3C: DataErrorStatus
1020 SMIx40: SciMap0

1021 SMIx44: SciMap1
1021 SMIx48: SciMap2
1022 SMIx4C: SciMap3
1022 SMIx50: SciMap4
1023 SMIx54: SciMap5
1023 SMIx58: SciMap6
1024 SMIx5C: SciMap7
1024 SMIx60: SciMap8
1024 SMIx64: SciMap9
1024 SMIx68: SciMap10
1025 SMIx6C: SciMap11
1025 SMIx70: SciMap12
1026 SMIx74: SciMap13
1026 SMIx78: SciMap14
1027 SMIx7C: SciMap15
1027 SMIx80: SmiStatus0
1028 SMIx84: SmiStatus1
1029 SMIx88: SmiStatus2
1030 SMIx8C: SmiStatus3
1030 SMIx90: SmiStatus4
1031 SMIx94: SmiPointer
1031 SMIx96: SmiTimer
1031 SMIx98: SmiTrig0
1032 SMIx9C: SmiTrig1
1033 SMIxA0: SmiControl0
1034 SMIxA4: SmiControl1
1034 SMIxA8: SmiControl2
1035 SMIxAC: SmiControl3
1036 SMIxB0: SmiControl4
1036 SMIxB4: SmiControl5
1037 SMIxB8: SmiControl6
1037 SMIxBC: SmiControl7
1037 SMIxC0: SmiControl8
1038 SMIxC4: SmiControl9
1038 SMIxC8: IoTrapping0
1038 SMIxD0: MemTrapping0
1039 SMIxD4: MemRdOvrData0
1039 SMIxF0: CfgTrapping0
1040 WDTx00: WatchdogControl
1040 WDTx04: WatchdogCount
1041 AcDcTimerx00: AcTimerValue
1041 AcDcTimerx04: AcExpiredTimerPolicy
1041 AcDcTimerx08: AcTimerStatus
1042 AcDcTimerx10: DcTimerValue
1042 AcDcTimerx14: DcExpiredTimerPolicy
1042 AcDcTimerx18: DcTimerStatus
1042 AcDcTimerx20: AcDcTimerCtrl
1044 AL2AHBx00: Hard Address Low
1044 AL2AHBx04: Hard Address High
1044 AL2AHBx10: Control
1044 AL2AHBx14: Status
1044 AL2AHBx18: Mask
1045 AL2AHBx20: HClk Hard Address Low
1045 AL2AHBx24: HClk Hard Address High
1045 AL2AHBx30: HClk Control
1045 AL2AHBx34: HClk Status
1045 AL2AHBx38: HClk Mask
1045 AL2AHBx40: AMBA Control 1
1046 AL2AHBx44: AMBA Control 2
1046 AL2AHBx50: AMBA Status 1
1047 AL2AHBx54: AMBA Status 2
1047 AL2AHBx60: AMBA Mask1
1047 AL2AHBx64: AMBA Mask2
1049 DMA[1:0]x000: DMA Manager Status Register (DSR)
1049 DMA[1:0]x004: DMA Program Counter Register(DPC)
1049 DMA[1:0]x020: Interrupt Enable Register (INTEN)
1050 DMA[1:0]x024: Event-Interrupt Raw Status Register (INT_EVENT_RIS)
1050 DMA[1:0]x028: Interrupt Status Register (INTMIS)
1050 DMA[1:0]x02C: Interrupt Clear Register (INTCLR)
1050 DMA[1:0]x030: Fault Status DMA Manager Register (FSRD)
1051 DMA[1:0]x034: Fault Status DMA Channel Register (FSRC)
1051 DMA[1:0]x038: Fault Type DMA Manager Register (FTRD)
1052 DMA[1:0]x0[5C:40:Step4]: Fault Type Register
1054 DMA[1:0]x1[08:00:Step8]: Channel [1:0] Status Register (CSR[1:0])
1055 DMA[1:0]x1[0C:04:Step8]: Channel [1:0] Program Counter Register (CPC[1:0])
1055 DMA[1:0]x4[20:00:Step32]: Channel [1:0] Source Address Register (SAR[1:0])
1055 DMA[1:0]x4[24:04:Step32]: Channel [1:0] Destination Address Register (DAR[1:0])
1055 DMA[1:0]x4[28:08:Step32]: Channel [1:0] Control Register (CCR[1:0])
1057 DMA[1:0]x4[2C:0C:Step32]: Channel [1:0] Loop Counter 0 Register (LC0[1:0])
1058 DMA[1:0]x4[30:10:Step32]: Channel [1:0] Loop Counter 1 Register (LC1[1:0])
1058 DMA[1:0]xD00: Debug Status Register (DBGSTATUS)
1058 DMA[1:0]xD04: Debug Command Register (DBGCMD)
1058 DMA[1:0]xD08: Debug Instruction 0 Register (DBGINST0)
1058 DMA[1:0]xD0C: Debug Instruction 1 Register (DBGINST1)
1059 DMA[1:0]xE00: Configuration Register 0 (CR0)
1060 DMA[1:0]xE04: Configuration Register 1 (CR1)
1060 DMA[1:0]xE08: Configuration Register 2 (CR2)
1060 DMA[1:0]xE0C: Configuration Register 3 (CR3)
1060 DMA[1:0]xE10: Configuration Register 4 (CR4)
1061 DMA[1:0]xE14: DMA Configuration Register (CRD)
1062 DMA[1:0]xE80: Watchdog Register (WD)
1062 DMA[1:0]xFE0: Peripheral Identification Register 0
1062 DMA[1:0]xFE4: Peripheral Identification Register 1
1062 DMA[1:0]xFE8: Peripheral Identification Register 2
1062 DMA[1:0]xFEC: Peripheral Identification Register 3
1062 DMA[1:0]xFF0: Component Identification Register 0
1063 DMA[1:0]xFF4: Component Identification Register 1
1063 DMA[1:0]xFF8: Component Identification Register 2
1063 DMA[1:0]xFFC: Component Identification Register 3
1064 I2C[3:0]x00: I2C Control Register (IC_CON)
1065 I2C[3:0]x04: I2C Target Address Register (IC_TAR)
1066 I2C[3:0]x08: I2C Slave Address Register (IC_SAR)
1066 I2C[3:0]x10: I2C RX/TX Data Buffer and Command Register (IC_DATA_CMD)
1067 I2C[3:0]x14: Standard Speed I2C Clock SCL High Count Register (IC_SS_SCL_HCNT)
1067 I2C[3:0]x18: Standard Speed I2C Clock SCL Low Count Register (IC_SS_SCL_LCNT)
1067 I2C[3:0]x1C: Fast Speed I2C Clock SCL High Count Register (IC_FS_SCL_HCNT)
1067 I2C[3:0]x20: Fast Speed I2C Clock SCL Low Count Register (IC_FS_SCL_LCNT)
1068 I2C[3:0]x24: High Speed I2C Clock SCL High Count Register (IC_HS_SCL_HCNT)
1068 I2C[3:0]x28: High Speed I2C Clock SCL Low Count Register (IC_HS_SCL_LCNT)
1068 I2C[3:0]x2C: I2C Interrupt Status Register (IC_INTR_STAT)
1069 I2C[3:0]x30: I2C Interrupt Mask Register (IC_INTR_MASK)
1069 I2C[3:0]x34: I2C Raw Interrupt Status Register (IC_RAW_INTR_STAT)

1070 I2C[3:0]x38: I2C Receive FIFO Threshold Register (IC_RX_TL)
1071 I2C[3:0]x3C: I2C Transmit FIFO Threshold Register (IC_TX_TL)
1071 I2C[3:0]x40: Clear Combined and Individual Interrupt Register (IC_CLR_INTR)
1071 I2C[3:0]x44: Clear RxUnder Interrupt Register (IC_CLR_RX_UNDER)
1071 I2C[3:0]x48: Clear RxOver Interrupt Register (IC_CLR_RX_OVER)
1071 I2C[3:0]x4C: Clear TxOver Interrupt Register (IC_CLR_TX_OVER)
1072 I2C[3:0]x50: Clear RdReq Interrupt Register (IC_CLR_RD_REQ)
1072 I2C[3:0]x54: Clear TxAbrt Interrupt Register (IC_CLR_TX_ABRT)
1072 I2C[3:0]x58: Clear RxDone Interrupt Register (IC_CLR_RX_DONE)
1072 I2C[3:0]x5C: Clear Activity Interrupt Register (IC_CLR_ACTIVITY)
1072 I2C[3:0]x60: Clear StopDet Interrupt Register (IC_CLR_STOP_DET)
1073 I2C[3:0]x64: Clear StartDet Interrupt Register (IC_CLR_START_DET)
1073 I2C[3:0]x68: Clear GenCall Interrupt Register (IC_CLR_GEN_CALL)
1073 I2C[3:0]x6C: I2C Enable Register (IC_ENABLE)
1073 I2C[3:0]x70: I2C Status Register (IC_STATUS)
1074 I2C[3:0]x74: I2C Transmit FIFO Level Register (IC_TXFLR)
1074 I2C[3:0]x78: I2C Receive FIFO Level Register (IC_RXFLR)
1075 I2C[3:0]x7C: I2C SDA Hold Time Length Register (IC_SDA_HOLD)
1075 I2C[3:0]x80: I2C Transmit Abort Source Register (IC_TX_ABRT_SOURCE)
1076 I2C[3:0]x94: I2C SDA Setup Register (IC_SDA_SETUP)
1077 I2C[3:0]x98: I2C ACK General Call Register (IC_ACK_GENERAL_CALL)
1077 I2C[3:0]x9C: I2C Enable Status Register (IC_ENABLE_STATUS)
1078 I2C[3:0]xA0: I2C SS and FS Spike Suppression Limit Register (IC_FS_SPKLEN)
1078 I2C[3:0]xA4: I2C HS Spike Suppression Limit Register (IC_HS_SPKLEN)
1078 I2C[3:0]xF4: Component Parameter 1 Register (IC_COMP_PARAM_1)
1079 I2C[3:0]xF8: I2C Component Version Register (IC_COMP_VERSION)
1079 I2C[3:0]xFC: I2C Component Type Register (IC_COMP_TYPE)
1080 UART[1,0]x00: DataBuffer and Divisor Latch Low
1080 UART[1,0]x04: Divisor Latch High (DLH)
1081 UART[1,0]x04: Interrupt Enable Register (IER)
1081 UART[1,0]x08: Interrupt Identification Register (IIR)
1082 UART[1,0]x08: FIFO Control Register (FCR)
1083 UART[1,0]x0C: Line Control Register (LCR)
1083 UART[1,0]x10: Modem Control Register (MCR)
1084 UART[1,0]x14: Line Status Register (LSR)
1084 UART[1,0]x18: Modem Status Register (MSR)
1085 UART[1,0]x1C: Scratchpad Register (SCR)
1086 UART[1,0]x[64:30:Step4]: Shadow Receive Buffer Register (SRBR)
1086 UART[1,0]x[6C:30:Step4]: Shadow Transmit Holding Register (STHR)
1086 UART[1,0]x70: FIFO Access Register (FAR)
1086 UART[1,0]x74: Transmit FIFO Read (TFR)
1086 UART[1,0]x78: Receive FIFO Write (RFW)
1087 UART[1,0]x7C: UART Status Register (USR)
1087 UART[1,0]x80: Transmit FIFO Level (TFL)
1087 UART[1,0]x84: Receive FIFO Level (RFL)
1088 UART[1,0]x88: Software Reset Register (SRR)
1088 UART[1,0]x8C: Shadow Request To Send (SRTS)
1088 UART[1,0]x90: Shadow Break Control Register (SBCR)
1088 UART[1,0]x94: Shadow DMA Mode (SDMAM)
1088 UART[1,0]x98: Shadow FIFO Enable (SFE)
1089 UART[1,0]x9C: Shadow RCVR Trigger (SRT)
1089 UART[1,0]xA0: Shadow TX Empty Trigger (STET)
1089 UART[1,0]xA4: Halt TX (HTX)
1089 UART[1,0]xA8: DMA Software Acknowledge (DMASA)
1089 UART[1,0]xF4: Component Parameter Register (CPR)
1090 UART[1,0]xF8: UART Component Version (UCV)
1090 UART[1,0]xFC: Component Type Register (CTR)
1090 UART[1,0]x400: Rx Control Register (RxCtrl)
1091 UART[1,0]x404: Rx Transfer Count Register(RxTransCount)
1091 UART[1,0]x410: Tx Control Register (TxCtrl)
1091 UART[1,0]x414: Tx Transfer Count Register(TxTransCount)
1091 UART[1,0]x420: DmaIrq Register (DmaIrq)